MAY 1977 VOLUME 2, Number 5 $1.50 in USA

BYTE

the small systems journal



Meet the most powerful μC system available for dedicated work. Yet it's only $595.*

Here's the muscle you've been telling us you wanted: a powerful Cromemco microcomputer in a style and price range ideal for your dedicated computer jobs—ideal for industrial, business, instrumentation and similar applications.

It's the new Cromemco Z-2 Computer System. Here's some of what you get in the Z-2 for only $595:

- The industry's fastest μP board (Cromemco's highly regarded 4 MHz, 250-nanosecond cycle time board).
- The power and convenience of the well-known Z-80 μP.
- A power supply you won't believe (+8V @ 30A, +18V and −18V @ 15A — ample power for additional peripherals such as floppy disk drives).
- A full-length *shielded* motherboard with 21 card slots.
- Power-on-jump circuitry to begin automatic program execution when power is turned on.
- S-100 bus.
- Standard rack-mount style construction.
- All-metal chassis and dust case.
- 110- or 220-volt operation.

DEDICATED APPLICATIONS

The new Z-2 is specifically designed as a powerful but economical dedicated computer for systems work. Notice that the front panel is entirely free of controls or switches of any kind. That makes the Z-2 virtually tamper-proof. No accidental program changes or surprise memory erasures.

FASTEST, MOST POWERFUL μC

Cromemco's microcomputers are the fastest and most powerful available. They use the Z-80 microprocessor which is widely regarded as the standard of the future. So you're in the technical fore with the Z-2.

BROAD SOFTWARE/PERIPHERALS SUPPORT

Since the Z-2 uses the Z-80, your present 8080 software can be used with the Z-2. Also, Cromemco offers broad software support including a monitor, assembler, and a BASIC interpreter.

The Z-2 uses the S-100 bus which is supported by the peripherals of dozens of manufacturers. Naturally, all Cromemco peripherals such as our 7-channel A/D and D/A converter, our well-known BYTESAVER with its built-in PROM programmer, our color graphics interface, etc., will also plug into the S-100 bus.

LOW, LOW PRICE

You'll be impressed with the Z-2's low price, technical excellence and quality. So see it right away at your computer store—or order directly from the factory.

Z-2 COMPUTER SYSTEM KIT (MODEL Z-2K) (includes 4 MHz μP card, full-length 21-card-slot motherboard, power supply, one card socket and card-guide set, and front panel; for rack mounting) . . . .$595.

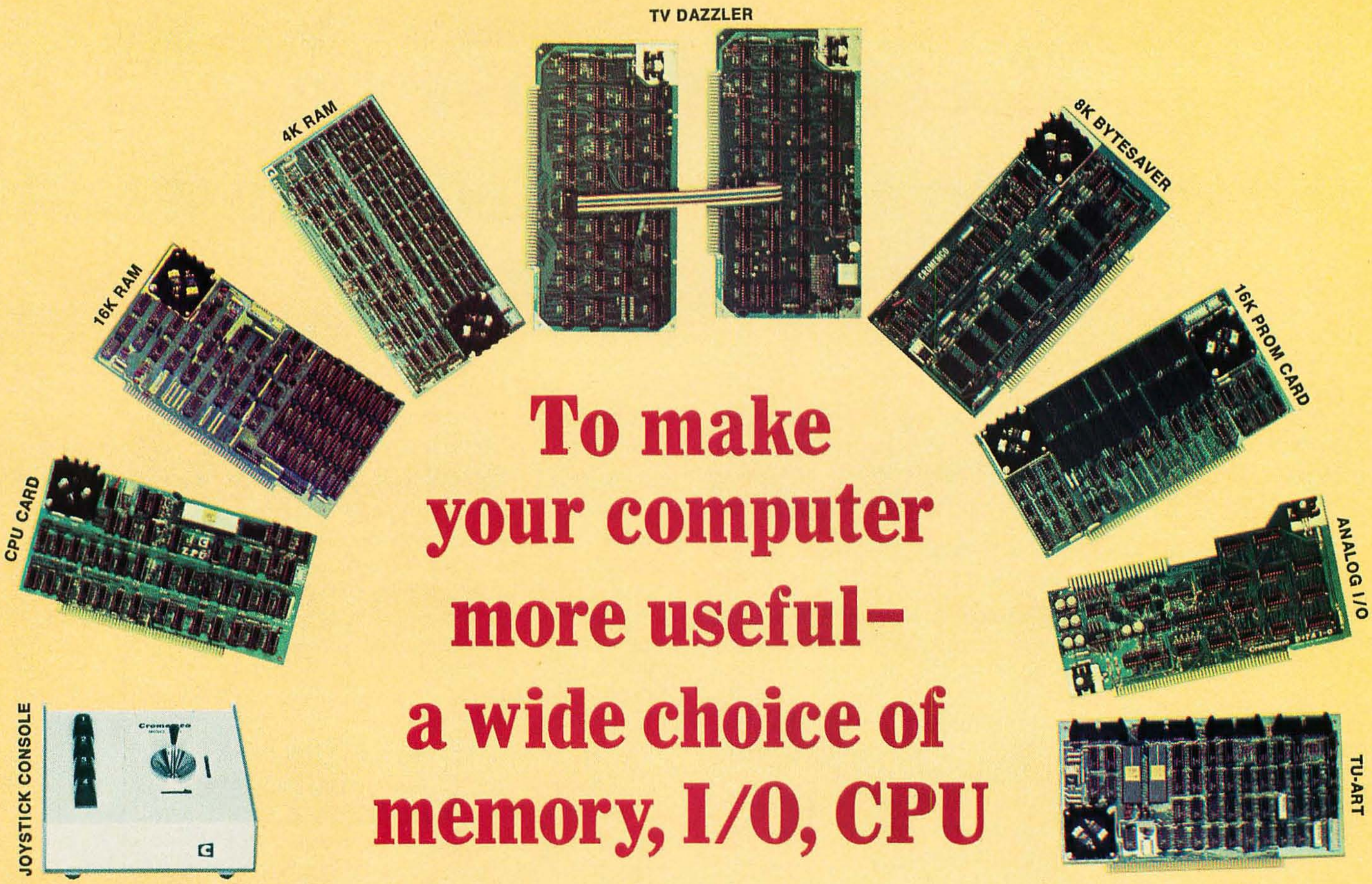
Z-2 COMPUTER SYSTEM ASSEMBLED (MODEL Z-2W) (includes the above as well as all 21 sockets and card guides and a cooling fan; for rack mounting) . . .$995.



Shown with optional bench cabinet

***kit price**

Cromemco
i n c o r p o r a t e d
Specialists in computers and peripherals
2432 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400



To make your computer more useful– a wide choice of memory, I/O, CPU

Your computer's usefulness depends on the capability of its CPU, memories, and I/O interfaces, right?

So here's a broad line of truly useful computer products that lets you do interesting things with your Cromemco Z-1 and Z-2 computers. And with your S-100-compatible Altairs and IMSAIs, too.

CPU

- Z-80 MICROPROCESSOR CARD. The most advanced μP card available. Forms the heart of our Z-1 and Z-2 systems. Also a direct replacement for Altair/IMSAI CPUs. Has 4-MHz clock rate and the power of the Z-80 μP chip. Kit (Model ZPU-K): $295. Assembled (Model ZPU-W): $395.

MEMORIES

- 16K RAM. The fastest available. Also has bank-select feature. Kit (Model 16KZ-K): $495. Assembled (Model 16KZ-W): $795.
- 4K RAM. Bank-select allows expansion to 8 banks of 64K bytes each. Kit (Model 4KZ-K): $195. Assembled (Model 4KZ-W): $295.
- THE BYTESAVER — an 8K capacity PROM card *with integral programmer.* Uses high-speed 2708 erasable PROMs. A must for all computers. Will load 8K BASIC into RAM in less than a second. Kit (Model BSK-0): $145. Assembled (Model BSW-0): $245.
- 16K CAPACITY PROM CARD. Capacity for up to 16K of high-speed 2708 erasable PROM. Kit (Model 16KPR-K): $145. Assembled (Model 16KPR-W): $245.

I/O INTERFACES

- FAST 7-CHANNEL DIGITAL-ANALOG I/O. Extremely useful board with 7 A/D channels and 7 D/A channels. Also one 8-bit parallel I/O channel. Kit (Model D + 7A-K): $145. Assembled (Model D + 7A-W): $245.
- TV DAZZLER. Color graphics interface. Lets you use color TV as full-color graphics terminal. Kit (Model CGI-K): $215. Assembled (Model CGI-W): $350.
- DIGITAL INTERFACE (OUR NEW TU-ART). Interfaces with teletype, CRT terminals, line printers, etc. Has not one but *two serial I/O ports* and *two 8-bit parallel I/O ports* as well as 10 on-board interval timers. Kit (Model TRT-K): $195. Assembled (Model TRT-W): $295.
- JOYSTICK. A console that lets you input physical position data with above Model D + 7 A/D card. For games, process control, etc. Contains speaker for sound effects. Kit (Model JS-1-K): $65. Assembled (Model JS-1-W): $95.

PROFESSIONAL QUALITY

You get first-class quality with Cromemco.

Here are actual quotes from articles by independent experts: "The Cromemco boards are absolutely beautiful" . . . "The BYTESAVER is tremendous" . . . "Construction of Cromemco I/O and joystick are outstanding" . . . "Cromemco peripherals ran with no trouble whatsoever."

Everyone agrees. Cromemco is tops.

STORES/MAIL

So count on Cromemco. Look into these Cromemco products at your store. Or order by mail from the factory.

We wish you pleasure and success with your computer.

Cromemco
i n c o r p o r a t e d
Specialists in computers and peripherals
2432 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

In the Queue

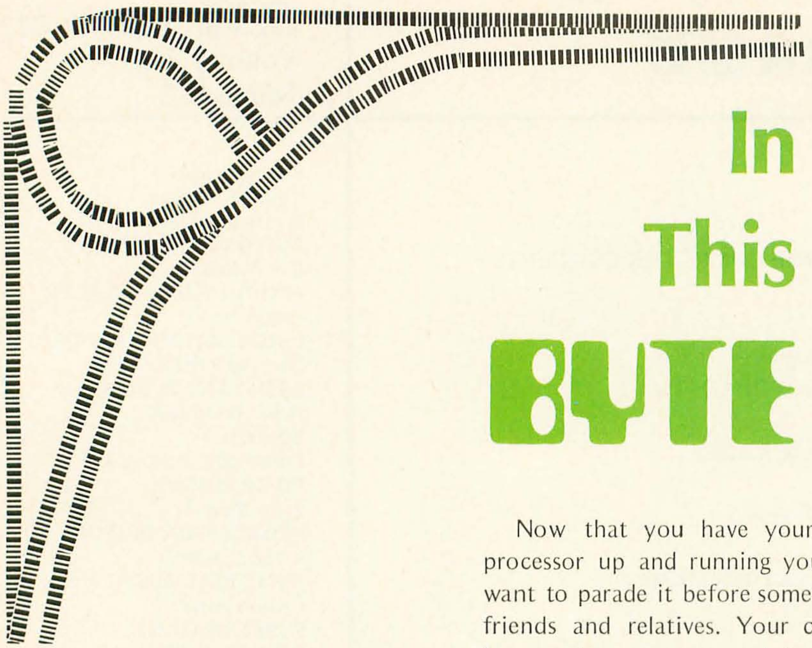
BYTE MAY 1977
Volume 2
Number 5

Foreground

Background

Nucleus

BYTE is published monthly by BYTE Publications Inc, 70 Main St, Peterborough NH 03458. Address all mail except subscriptions to above address; phone (603) 924-7217. Address all editorial correspondence to the editor at the above address. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage. Not responsible for lost manuscripts or photos. Opinions expressed by the authors are not necessarily those of BYTE. Address all subscriptions, change of address, Form 3579, and fulfillment complaints to BYTE Subscriptions, PO Box 361, Arlington MA 02174; phone (617) 646-4329.
Second class postage paid at Peterborough NH 03458 and at additional mailing offices.
Subscriptions are $12 for one year, $22 for two years, and $32 for three years in the USA and its possessions. Add $5.50 per year for subscriptions to Canada and Mexico. For air delivery to western Europe, and for surface delivery worldwide, $25 for a one year subscription only. Worldwide air delivery available at additional rates upon request.
Single copy price is $1.50 in the USA and its possessions, $2 in Canada and Mexico, and $3 elsewhere.
Foreign subscriptions and sales should be remitted in United States funds. Printed in United States of America.



page 84

In This BYTE

About the Cover

Birgit Quednau, a student of biomedical technique at the university in Giessen, Germany, who is interested in small computer systems and batik, combined both fascinations to produce this month's cover, another winning entry in BYTE's Computer Art Contest.

Ideas and imagination are the inputs to creative uses of computers. Ted M Lau has set down some of his thoughts on potential personalized uses of computers in his **Catalog of Liberating Home Computer Concepts.** Some of his suggestions can be implemented with present technology. Some will have to await further development. In either case the prospects are exciting.

Artificial intelligence has intrigued people for many years. The possibility that computers may be able to "think on their own" is one of the recurring themes of science fiction. Artificial intelligence is starting to pervade reality, not the super thinking beasts of fiction, but the machines that perform tasks normally thought to require intelligence. In his article, **Artificial Intelligence, an Evolutionary Idea,** Michael Wimble describes one type of artificial intelligence technique which readers may find quite useful.

What does it take to make a computer system complete to the point of plugging it into the wall, plugging it into a color television, and turning it on? Stephen Wozniak of Apple Computer describes the design of such a system in his product description article on the Apple-II.

Now that you have your microprocessor up and running you surely want to parade it before some of your friends and relatives. Your cluttered basement or garage workshop, however, may not be the best place to demonstrate your brainchild. In his article, **Come Upstairs and Be Respectable,** Steve Ciarcia describes his solution to this dilemma by installing a remote keyboard and video monitor in his den. This arrangement is great for parties and other gatherings since the vital components of the processor cannot be subjected to the whims of some unknowledgeable person.

Joseph J Carr, in his first of a two part article on **Interfacing With an Analog World,** gives us an insight into transducers and some of the problems of processing their outputs into signals which can be digitized by an analog to digital converter.

Adding floating point calculation abilities to your microprocessor can represent a quantum leap forward in performance. The floating point functions discussed by Sheldon Linker in his article, **What's in a Floating Point Package?,** will allow you to perform addition, subtraction, multiplication and division with extremely large or small numbers, using software exclusively.

How can you convert one of those surplus keyboards without encoders into a usable text input? Find out by reading Bob Brehm's **Using a Keyboard ROM.**

Jack Emmerichs' Tiny Assembler presentation is completed in the second part of his article in this issue: **Implementing the Tiny Assembler.** Readers will find a discussion of structured code details for a 6800 as well as complete object code and sufficient information to get Tiny Assembler 6800 up and running in any machine with MIKBUG and at least 4 K of memory starting at address 0000.

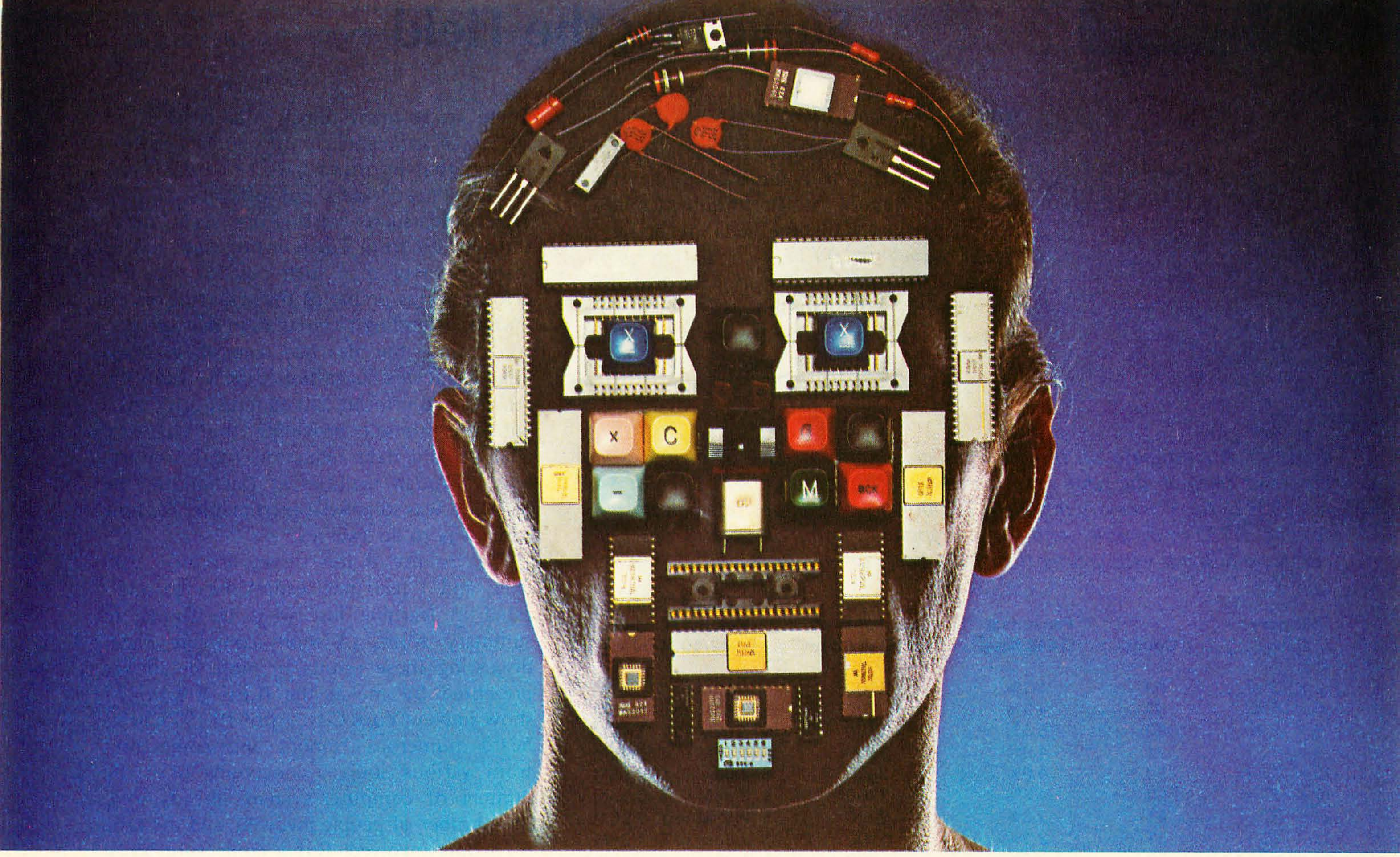
Last month Michael S McNatt described the various types of Baudot teleprinters that are available on the surplus market today. This month he describes various ways in which these devices can be interfaced to a microprocessor using both hardware and software techniques in his article, **A Guide to Baudot Machines: Part 2, Interfacing Techniques.**

Using someone else's interpreter or compiler is the normal mode of operation for anyone contemplating a high level language. But with any complicated piece of software, use sometimes requires a bit of ingenuity on the part of the user as David Chapman points out in his description of a nit in many a BASIC interpreter. Turn to **All This Just to Print a Quotation Mark?** You'll also find a short glossary including some very important terms in the world of applications software.

The 8080 microprocessor generally performs operations using 8 bit words. There are, however, several ways in which 16 bit words can be manipulated on the 8080. In the article, **8080 Programming Notes,** John McCarty and Paul Krystosek elucidate on 16 bit data manipulation.

"A mess of parts doth not a computer make"

ANON



Undoubtedly ANON was talking about some of today's PC computers. Exciting as they may be, most of them give you a few surprises you really don't need.

For example you have the board all wired, and suddenly it occurs to you, "Where's the TV interface?"

Surprise! (The TV interface is extra and costs $200 plus.)

"And the keyboard interface — where's that?"

Surprise! (Add another $185 extra.)

"How about an audio cassette interface so you can store and enter low cost programs and large amounts of data?"

Surprise! (Add $175 more.)

"And what about software?"

Surprise! (Each program will cost you $20 plus.)

See what we mean?

The new Sol PC Terminal Computer. It's a complete kit, hardware, software and no surprises for $475.

It's the first small computer conceived as part of a total system.

Not only do you get everything you need, but our systems approach is more economical, too. The $475 kit (fully assembled and tested, $745) includes all of the following as standard features:

Video display interface.
Keyboard interface.
Audio cassette interface.
8080 microprocessor.
512 eight bit bytes of PROM on a plug-in personality module.
2048 eight-bit words of RAM.
A 1024 character video display generator.
Serial and parallel interfaces for connection to external devices and an edge connector for memory expansion.

Optional extras include a power supply, video monitor, ASCII keyboard, walnut case, floppy disk system, high speed paper tape reader, PROM programmer, and color graphics interface. We also have a library of low cost software that's the best and most extensive in the industry.

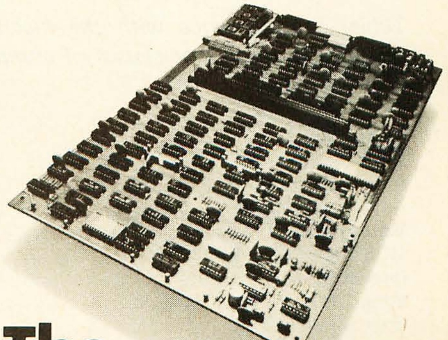
In short, you get a complete, powerful, small computer and smart terminal that's fully expandable to a larger Sol system, as your requirements grow.

Surprises are great at Christmastime. But when you're opening the box of your new computer kit, you want everything you need to be there.

And with Sol, it is.

To order or to get more information write Processor Technology, 6200 Hollis St., Box B, Emeryville, CA 94608. (415) 652-8080.

Processor Technology Corporation



The Sol PC Terminal Computer

Editorial

Surveying the Field

By Carl Helmers

Who Uses Personal Computers?

In the fall of 1976 Mike O'Brian, who is a second year student in the MBA program of the Tuck School at Dartmouth College, approached us with the idea of doing a reader survey of BYTE's circulation as an independent study project. For those among our readers who contemplate starting a magazine (or any other business for that matter) you should be aware of one fact: There are so many things to be done which are never anticipated until you get into the project that priorities have to be made and various "neat" ideas put off on occasion. This reader survey task is one such "neat" thing to do from our point of view, but of lower priority than such obvious high priority items as preparing the magazine once a month, attending to circulation matters, etc. As a result, the idea had languished. When Mike made his suggestion that we fund the mailing and printing and administrative costs with he himself providing the statistical analysis and data reduction for his course work through facilities at Dartmouth, the response was somewhat predictable: We now had a way to do a survey.

The results of this survey are now available, and I feel that readers may be interested in learning just what we found, in summary form. (A more complete paper discussing these results in more detail was presented by me at the IEEE Electro 77 show in New York City, April 19 1977.) I've heard numerous rumors and conjectures from various sources concerning both the number of computer systems already sold, the number of people involved, and the type of people involved. But a conjecture based on a personal assessment of the marketplace can sometimes be no better than an arbitrary assertion. One way to change such assertions into a better estimate of reality is to make empirical measurements of the data such as we have performed. With this spirit of measurement and observation, Mike and I set out to design and implement a survey of the readers of BYTE, circa October-November 1976.

The basic idea of a survey is to choose a subset of the total population of readers, then mail them a detailed questionnaire to characterize various aspects of their participation in the field of personal computing. Then, since we presume that the sampled population is typical of the total population, we can generalize the results to the entire set of BYTE readers. Mike and I designed a 4 page detailed questionnaire covering numerous aspects of the personal computing field, which was sent to an "every nth" sample of the BYTE circulation list as it stood the first week of October 1976. By selecting only those subscribers who had paid in full, and applying an every nth name algorithm, we printed a list of 2163 subscribers who would receive the survey. The survey questionnaire,

Table 1: An estimate of the total dollar volume to date of memory and processor sales at the retail level, based upon a typical current price of $42 per 1024 bytes of memory installed, 78,000 current demonstrated enthusiasts reading BYTE magazine, and an extrapolated total of 26,100 operating computer systems with greater than 1024 bytes of memory.

Memory Range	Count	Percentage (1448 base)	Extrapolated Installed Count	Installed Dollar Volume Range (@ $42/1024 bytes)
1 K–2 K	59	4.1%	3200	$ .13M to $ .27M
3 K–4 K	62	4.3%	3300	$ .42M to $ .55M
5 K–8 K	80	5.5%	4300	$ .90M to $ 1.45M
9 K–12 K	71	4.9%	3800	$ 1.44M to $ 1.92M
13 K–16 K	70	4.9%	3800	$ 2.08M to $ 2.56M
17 K–24 K	62	4.3%	3300	$ 2.36M to $ 3.33M
25 K–32 K	41	2.8%	2200	$ 2.31M to $ 2.96M
33 K–65 K	31	2.1%	1700	$ 2.36M to $ 4.64M
66 K and up	9	0.6%	500	$ 1.39M to $ 1.39M*
Total owners of operating computer systems with 1 K or more bytes of programmable memory			26,100	$13.4M to $19.1M
Allowing a conservative range of $1000 to $1500 for processor, video interface, keyboard and other peripherals at the retail level				$26.1M to $39.2M
Estimated total dollar volume to date				$39.5M to $58.3M

*We have no upper bound on this range, so the lower bound is used for these estimates.

Table 2: Experience with the machine language or assembly language of specific computers or classes of computers. Percentages calculated on a 1448 response base.

Category	Counts	Percent
Any large computer	739	51%
Any minicomputer	849	59%
Motorola 6800	338	23%
MOS Technology 6502	163	11%
Intel 8080	607	42%
Zilog Z-80	114	8%
Intersil IM6100	67	5%
Digital Equipment LSI-11	185	13%
RCA 1802	59	4%
Signetics 2650	35	2%
National PACE	62	4%

a prepaid postal reply envelope, and a cover letter were sent to each of these readers. The cover letter also contained an entry blank for an incentive contest with five life subscriptions to BYTE awarded.

The survey forms were kept separate from the entry blanks to preserve anonymity of the respondents, and the forms were sent to Mike at Dartmouth to be keystroked and analyzed. The response rate to this survey was nothing short of fantastic, according to Mike's marketing advisors at Dartmouth. 1448 of the 2163 people surveyed returned forms by the November 13 1976 deadline. This is a 67% response.

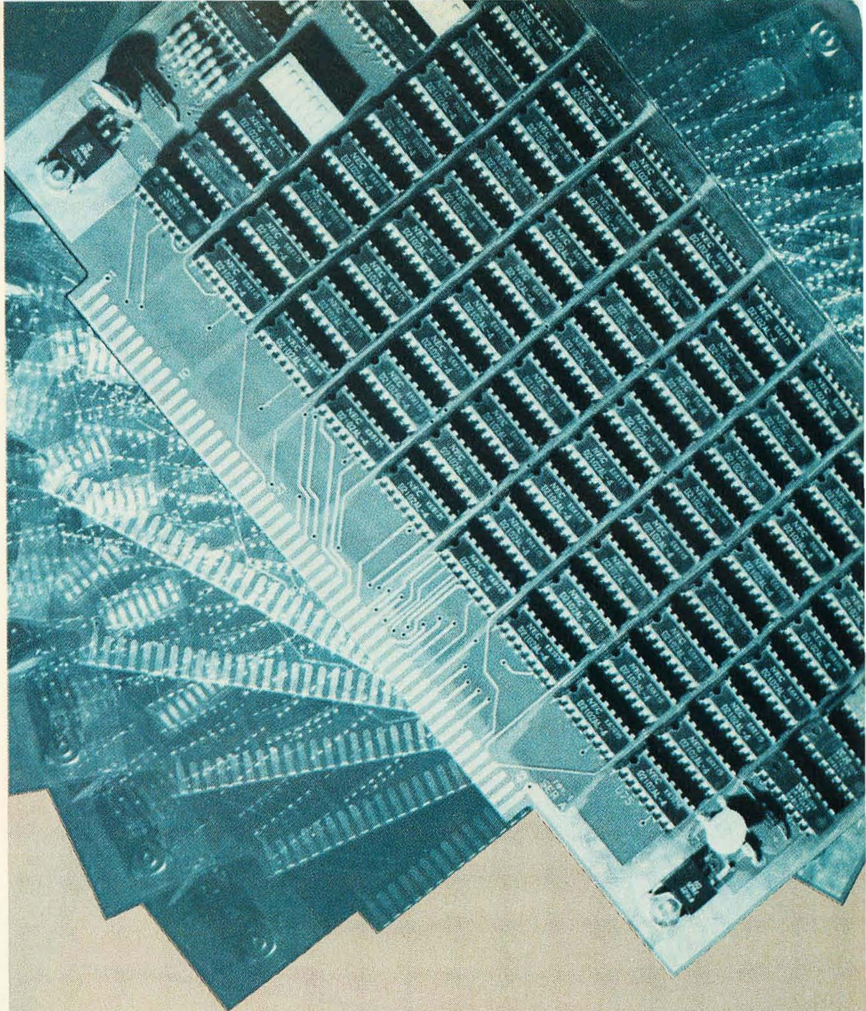
There are numerous ways to describe characteristics of individuals, based on various biological, educational and economic factors. One parameter which is of interest in passing is the question of sex. It is fashionable in this era to minimize differences in interests, tastes (and therefore, capabilities) among people based on biological gender. Popular illusions to the contrary, it is an observed data point in our survey that 99% of the computer enthusiasts are male, 1% female. (These percentages are based on those who answered the questions; those who ignored the question were at the 1% level also.) This draws no conclusions concerning the reasons for the skew relative to the total population distribution among human beings on this planet, but it is an interesting point of fact.

Another question often asked is that of educational level. What kind of intellectual capabilities are demonstrated by those who get into the computer passion as readers of BYTE and practitioners of these arts? We asked readers what was the highest level of formal education attained or actively in progress. The following table summarizes the responses found:

High school	11%
Junior college	15%
Bachelor's degree	41%
Master's degree	22%
PhD	9%
MD or DDS	2%
— — — —	
No answer to query	1%

This shows a definite concentration of people with a significantly high educational background, which of course is consistent with the economic question "What is your income?" which obtained the following results:

$ 0 to $10k	8%
$10k to $15k	19%
$15k to $20k	22%
$20k to $25k	16%



Now. THE PERFECT MICROCOMPUTER

VECTOR 1



It's a two touch affair

VECTOR 1 is based on the 8080A Microprocessor and S-100 Bus -

EASY TO ASSEMBLE, EASY TO ACCESSORIZE, EASY TO BUY AND VERY EASY TO OPERATE.

VECTOR GRAPHIC INC. T.M.
717 LAKEFIELD ROAD, SUITE F
WESTLAKE VILLAGE, CA 91361 • (805) 497-0733

$25k to $30k	15%
$30k and up	19%

The median income of this sample is $20k per annum, leading to the conclusion that computer hackers tend to be fairly well off economically.

What Have People Invested So Far?

Buying a computer system is in some respects like buying a house or a car. It is a long term capital investment by the individual, but it does have characteristics unlike the house or the car in its relative freedom from maintenance due to its preponderance of integrated circuits. In designing the survey, we made one mistake in the form's layout which prevented an easy tabulation of answers in the "How much have you spent on X?" category where X is some item like a floppy disk, main frame, terminal, etc. But we did get useful data on a key item which enables us to deduce a lower limit on total sales of computer products to date. This item is the number of bytes of memory contained in systems. We found that 35% of the readers had operating computer systems, based on one question. Correlating the responses to the question "How much memory do you have in thousands (K) of bytes?" with those readers who had operating computer systems, we found the distribution of table 1. This table also contains an extrapolated figure based upon the February 1 1977 BYTE circulation of 78,000 readers, and an estimated dollar volume of sales of memory

Continued on page 155

Table 3: Language and programming experience and intentions. All percentages based on 1448 respondents.

	Frq	Occ	Lit	Int	None	Excl
Machine Language	26%	40%	66%	14%	17%	3%
Assembly Language	31%	34%	65%	18%	13%	3%
BASIC	21%	38%	59%	25%	14%	2%
FORTRAN	26%	43%	69%	11%	18%	2%
PL/I	6%	13%	19%	19%	54%	8%
ALGOL	3%	13%	16%	10%	66%	8%
COBOL	8%	15%	23%	11%	58%	8%
APL	3%	10%	13%	20%	59%	8%
RPG	2%	8%	10%	9%	73%	8%
Ded Appl Lang	12%	19%	31%	12%	48%	9%

Definitions:

Frq = frequent use, currently.
Occ = occasional use, or past use frequently.
Lit = literacy = Frq + Occ (ie: know language).
Int = intend to learn language.
None = no interest in language.
Excl = exclusions, did not answer question.

Letters

NOSTALGIA OF COMPUTING

The letter from Don Nyre in your February 1977 issue suggesting that we do more to preserve our computer heritage by preserving antique computers and publishing more articles on computer nostalgia and antique computers caught my eye. We here at Western Washington State College are doing our bit (?) to preserve our computer heritage by still running an IBM 7090 and cannibalizing a 7094 for parts. Yours truly cut his eye teeth on a Royal McBee LGP-30 and was weaned to an IBM 650 and from thence to an IBM 704 in graduate school.

Our 7090 is interfaced to our IBM 360/40, which is interfaced to one of our Interdata 7/32 minicomputers, which will shortly be interfaced to a MITS 680b. I suppose we should complete the circle by having the 680b push the START button on the 7090 console. In Seattle 709 and 7090 equipment is still the basis of Herb Burke's Puget Sound Computer Service. Interested readers are invited to ship their spare IBM 700 and 7000 series computers to Herb, COD. I am sure he will appreciate it.

Keep up the good work with BYTE.

Melvin Davidson
Curator, Computer Museum
Western Washington State College
Bellingham WA 98225

ANYONE HAVE A 6800 APL?

Include me in your list of BYTE readers interested in the APL programming language. I would gladly buy up a product that made it possible to run APL programs on my 6800 system.

Jay Voral
279 Twinlakes Dr
Sunnyvale CA 94086

ATTENTION AUTHORS: INFORMATION NEEDED

I wish your magazine and others like it would give more coverage to commercial equipment and software, and how those can be interfaced with the home market products you primarily focus on. The reason is that much of what you do cover appears to be useful for the same applications businesses and professional institutions use their computers for, but at astonishingly lower costs. Yet it remains puzzling how this home market equipment can be used with commercial products.

For example, the IBM 5100 desk top computer appears equivalent in capacity to the Altair and similar systems for the home; yet I, at least, find no mention of it in BYTE. */See BYTE December 1975 for a new product item on the IBM 5100 . . .* CH/ At the same time, very sophisticated software is marketed for the IBM 5100, such as applications packages in APL. Can any of that software be used by the amateur computer owner?

Another example is cassette recorders. I use some remote terminals for timesharing for which there are cassette recorders typically costing $3,000. These are the only recorders the terminal suppliers provide for their terminals. Is there any way all those cassette devices covered in BYTE can be used with the commercial grade remote terminals, which are themselves special purpose microprocessors (eg: GE Terminettes)?

A related example is modems. You have covered a variety of remote tape data transceiver devices, but never mention specifically what kind is required to operate by telephone a commercial timesharing facility as opposed to a user's own custom designed system.

George Lyons
280 Henderson St
Jersey City NJ 07302

APL in microland is alive and well; we know of several interpreters presently under development, and we are preparing a number of articles on the subject of implementing and using APL. With an APL interpreter for an 8080, 6800, 9900 or other processor, it will be possible to run public domain APL applications packages, of course allowing for detail differences between interpreters which will always exist. Your other queries might make good starting points for articles; so, authors take note.

ADDENDUMS TO THE F8 REVIEW

I would like to make a few additions to the article on the Fairchild F8 (February 1977, page 88). As correctly pointed out by Mr Baker, the 3851 program storage unit (PSU) is rather expensive for the hobbyist. But Fairchild now makes a 3861 peripheral input output (PIO) chip. The PIO is identical to the PSU except it does not contain the 1 K of ROM. Five versions of the 3861 are available, each mask programmed with different IO port addresses and interrupt vectors. This chip allows for adding IO ports to a system without adding more ROM. A 3861 is about half the price of a 3851. Thus using a 3850, 1 to 4 3861s, a 3852 DMI or 3853 SMI and PROM or programmable memory, one can build a complete system without having a 3851 mask programmed. We have done that here in a number of projects with much success.

Fairchild also makes a 3851 programmed with a debug program (called Fairbug) and some Teletype IO subroutines (part number 3851A). Fairbug allows one to read and write memory and the 64 scratchpad registers and load and dump programs, as well as execute a program. The IO subroutines do all the serializing and timing in software. Two bits of an IO port on the 3851A are used for serial input and serial output respectively. Two other bits are used to select the bps rate (110 or 300). This makes it very easy to hook up a terminal to the F8. No UART or bps rate generator is needed.

I think you are publishing an excellent magazine. Each month BYTE gets bigger and the articles get better. I'm interested in both hardware and software and you have a very good mix of both. I'm glad to hear that you will be running a series of articles on building a computer using the 6502. I have been using this processor in my home system since it first came out. I believe it has one of the best instruction sets of the available processors. I hope your articles will generate more interest in the 6502 so we can build up a large library of software for it.

Jeffry L Parker
Fairchild Systems Tech
1725 Technology Dr
San Jose CA 95110

STANDARD TAPE INTERFACES?

A lot of data which would be interesting to process with amateur computers is distributed at cost by the federal government in machine processible form (such as the population census). However, the data comes on half inch tape for mounting on a large-scale system. Is there any way to read these tapes into an Altair size system?

Jacob Schmidt
310 E 83 St
New York NY 10021

Yes: Build a controller to interface a surplus 7 or 9 track tape drive. This is a doable project which has already been successfully tackled by advanced home-brewers. Eventually we should see such controllers available on the market as specialty items; however, the cost of 7 or

The POLY 88 Microcomputer System

PolyMorphic Systems now offers the complete, assembled, personal computer system—the POLY 88 System 16. A full 16K system with high speed video display, alphanumeric keyboard, and cassette program storage. A BASIC software package providing the most advanced features available in the personal computing market. Features like PLOT and TIME, which utilize our video graphics and real-time clock. Others like VERIFY, so that you know your tape is good before you load another. Or input type-ahead so you can tell your program to run while the tape is still loading (it stores up to 64 characters of commands or question responses to be executed). All these plus a complete package of scientific functions, formatting options, and string capabilities. With the POLY 88 System 16 you can amaze your timesharing friends the very first night!

PolyMorphic Systems 11K BASIC

Size: 11K bytes.

Scientific Functions: Sine, cosine, log, exponential, square root, random number, x to the y power.

Formatted Output • Multi-line Function Definition • String Manipulation and String Functions • Real-Time Clock • Point-Plotting on Video Display • Array dimensions limited by memory • Cassette Save and Load of Names Programs • Multiple Statements per Line • Renumber • Memory Load and Store • 8080 Input and Output • If Then Else • Input type-ahead.

Commands: RUN, LIST, SCR, CLEAR, REN, CONT.

Statements: LET, IF, THEN, ELSE, FOR, NEXT, GOTO, ON, EXIT, STOP, END, REM, READ, DATA, RESTORE, INPUT, GOSUB, RETURN, PRINT, POKE, OUT.

Built in Functions: FREE, ABS, SGN, INT, LEN, CHR$, VAL, STR$, ASC, SIN, COS, RND, LOG, TIME, WAIT, EXP, SQRT, CALL, PEEK, INP, PLOT.

Systems Available. The POLY 88 is available in either kit or assembled form. It is suggested that kits be attempted only be persons familiar with digital circuitry. The following are two of the systems available.

System 2: is a kit consisting of the POLY 88 chassis, CPU, video circuit card, and cassette interface. Requires keyboard, TV monitor, and cassette recorder for operation. $690

System 16: consists of an assembled and tested System 2 with 16K of memory, keyboard, TV monitor, cassette recorder, 11K BASIC and Assembler on cassette tapes. $1995.

Prices and Specifications subject to change without notice.
California residents add 6% sales tax.

PolyMorphic Systems
460 Ward Drive Santa Barbara, Ca. 93111 (805) 967-2351



9 track IBM compatible drives is fairly high unless purchased via a surplus house.

6800 REGISTER TRANSFERS, CLARIFIED

Many readers must have spotted the goof in my letter, December 1976 BYTE, about the 6800. TAB and TBA *do* allow exchange between the two accumulators, and it is the X register that only communicates with A and B via memory. As the owner of a 6800 system, perhaps *you* should have spotted this! *[Maybe, but I can give the ultimate cop-out and say facetiously that it was your letter, after all. But the point is valid of course – and I do use TBA and TAB whenever necessary on my 6800 . .* CH*]*

H T Gordon
641 Paloma Av
Oakland CA 94610

PS: My disparaging comments on the IM6100 were based entirely on chapter 12 of volume 2 of the Osborne Associates book. Until then I knew nothing of the inner operation of the PDP-8 except that (through a FORTRAN compiler) it works fine. That this primitive hardware should survive (entirely on its accumulated software) was quite a shock! The Osborne book is a treasury of information on design. You should run an article on the *profound* differences among micros, and between minis and micros, to educate your readers who lack the time or background to plough through the Osborne book.

Old software eventually dies, but usually when the machines which use it have been scrapped. For example, not many people must be using the FORTRAN compiler for the SPIRAS-65 computer I saw auctioned in Cambridge MA in 1973 (or was it 1974?) for $85 (not bad for an 8 K 24 bit minicomputer processor in working condition, optimized with floating point for number crunching). As a second example, there are still very real programs still in operation on machines which could add, subtract, multiply, print, read cards, etc. I refer to all those old monster IBM tabulating equipments such as the 407, etc, which use plugboard programming to create algorithms of a simple order. Why throw out a working machine if it doesn't cost an amount approaching infinity to maintain it?

MEETING THE ECONOMICS OF STUDENTS

My main purpose in writing this letter is to voice a student's point of view. I feel that my view is representative of a number of college and high school students involved in computing across the country. Basically, I am of the opinion that we have been passed over by the industry. First of all, one must realize the limited budgets available to students. There are many powerful systems in existence but these are often too expensive for a student of meager earnings to afford. No matter how you slice it, a function software development system (processor 4 K bytes programmable memory minimum, PROM monitor, video display, cassette IO and packaging) costs well over $700, a conservative approximation. Students like myself don't have the money. Now I'm not saying that someone should offer a system like the one above for $200, since that would be ridiculous; my real complaint is that I have not seen a low cost alternative system which is designed with the following idea in mind: Most students have access to a dial up timesharing system, and could thus use a terminal of sorts that can be kept at home or in the dorm. So why not design the terminal with a processor in it, so that we can expand the terminal into a complete system a little bit at a time, while still using the thing as a terminal for the timesharing service? Basically, it would be like a Processor Technology SOL, but without the extra memory, cassette interface and the attractive but probably expensive case. If it were possible to make such a minimal terminal at a price around the $350 to $450 range, while retaining the ability to expand upwards, I think there would be a large market among students such as myself. There are great numbers of students throughout the nation who are capable of writing some excellent software for small systems, but have not even tried to get into this type of computing because of the cost involved.

Tony Aiuto
19 Old Field Ln
Great Neck NY

For the person with a tight budget, the single board computers and computer kits such as KIM-1, JOLT, the new Motorola 6800 evaluation kit, or the Intersil Intercept Jr (to name just a few) provide the least expensive starting point (minus timesharing options). All are expandable upwards but come sans video display, modem or keyboard. Or, if a finished high level language system is what you want, go to a department store and evaluate a programmable calculator. Alas, no video display in calculators, but toward the end of this year we can expect to see some interesting portable computing products which are variants of calculators scaled upwards (perhaps with video outputs) at prices in the calculator range which you can afford.

And then, of course, if you're a student at a college, there is the old tried and true method of locating a professor, selling him or her on the idea of getting

Complete Control.



Introducing IMSAI 8048 Single Board Control Computer.

Complete Control System

Intel developed the world's first single chip microcomputer. IMSAI has built the system to put it to work. Instantly!

Presenting the IMSAI Single Board Control Computer. A complete programmable computer and hardware control system on an 8½" x 10" board.

Just plug in the power source (a 5V power supply or a 6V battery will do), and you can be controlling your home environment today!

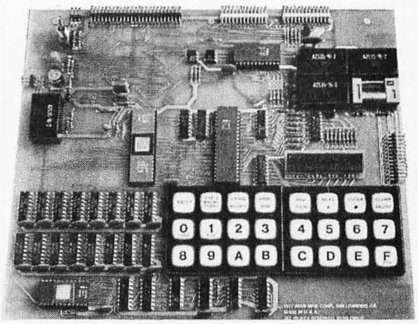
The Chip Designed for Control

Intel designed the 8048/8748 single chip microcomputer with one thought in mind. Complete control. Everything you need is there: CPU, RAM, 1K ROM/EROM, I/O, timer/counter, interrupts . . . the works.

The Board That Implements It Instantly

There's a 24 pad hexadecimal keyboard and 9-digit LED hex display already on board. So you can start controlling the coffeepot and the sprinklers the minute you get the IMSAI home. Without adding anything but the connecting wires.

You can run almost any peripheral available. Not to mention the kids' electric train. The IMSAI Control Computer is



RS232 compatible. There are 12 quasi-bidirectional I/O lines with handshaking, and 14 more regular I/O lines, 5 heavy duty relays, and Teletype and audio cassette interfaces. All on one board.

There's already an extra 1K of RAM on board, plus sockets for another 1K of RAM and 2K of ROM/EROM. Still need more memory? The IMSAI 8048 allows expansion up to 64K of RAM off board.

Ultimately, the only limit to this system is your imagination.

Now, that's control.

For instant control, use the coupon provided.

GENTLEMEN:
I want instant control!
☐ Send ROM Computer Kit $249*
☐ Send EROM Computer Kit $399
☐ Send assembled ROM Computer $299*
☐ Send assembled EROM Computer $499
☐ Send 5V power supply $99
*June delivery.
Check/MO enclosed. Amt. $ ______
Charge my: ☐ BAC ☐ M/C
______ Exp. Date ______
Sig. ______
☐ Send more information.
Name ______
Address ______
City ______
State/Zip ______

IMSAI®

IMSAI Manufacturing Corporation
14860 Wicks Blvd.
San Leandro, CA 94577
(415) 483-2093
TWX 910-366-7287

you and your school into personal computing studies, and having the school make the investment for you. This approach has been done by undergraduate and graduate students with minicomputers (and lately, microcomputers) for years.*

COVER KUDOS

It is truly refreshing to be reminded that art in America is not confined to museums. The choice of cover art for the January 1977 BYTE is a credit to your editorial staff. My highest praise goes out to Mr Tinney for his insight and well presented message. There is one small problem, however: All those people like me who removed the cover for framing are left with no protection for the first pages of the magazine.

Stuart Beatty
1424 34th St NW
Canton OH 44709

Magazines have often inspired excellent artwork, a practice which we at BYTE want to encourage. A classic example is the work that artist Maxfield Parrish performed early in his career illustrating magazine covers and stories early in this century. For those unfamiliar with the man, Maxfield Parrish had a unique and aesthetically pleasing style of painting which will probably class him among the great romantic artists of all time. When I first asked Robert Tinney to execute covers for BYTE (which were constrained to the cartoon format of early BYTES by our production means, not Robert's choice), I used Maxfield Parrish as represented in a book by Coy Ludwig as one of the mental models to be considered. The January cover was Robert's idea and came out quite well.

Are You Interested in Limited Edition Reprints of BYTE Covers?

We have received a significant number of requests for reprints of BYTE covers in full size poster form. However, the number is not sufficient to cover preparation and handling (as well as a bit of extra remuneration to the artists) at typical poster prices in the $2 to $5 range.

As a way of satisfying the demand for covers, we would like to see if there is reader interest in limited edition printings of covers (without the BYTE logo or lettering which is applied in the reduced magazine form of the paintings). A limited edition is defined as a run of $n \leqslant 100$ prints, approved by the artist, signed and numbered by the artist, and accompanied by a notarized affidavit certifying the limited quantity and position in the run. Such sales would be conducted as an auction, with the lowest numbers going to the highest bidders, ranked by the price bid. A minimum acceptable bid would be announced when the auction sale is announced.

If you are interested in bidding on such prints, let us know by sending a postcard listing the covers you'd like to see reprinted (by month and year), and the price range in which you would intend to bid in such an auction. Address the card to:

Cover Print Survey
BYTE Publications
70 Main St
Peterborough NH 03458

If we get enough response to warrant such reprints, we'll announce selections in a future issue and solicit bids. ■

COVER REPRINT INTEREST

Please let me know whether the front cover of February 1977 BYTE, or for that matter, any other of the computer art you publish, is going to be made available in poster-like form.

Richard H Delp
19703 Gary Av
Sunnyvale CA 94086

We have available presently one cover reprint poster: the July 1976 painting by Robert Tinney in its original form, which costs $2.95 postpaid. Order it through BITS (see advertisements in this issue).

FOR YOUR CANADIAN READERS' INFORMATION

Our store, The Computer Place, officially opened November 15 1976. Before that date, during renovations, we were selling micros and peripherals on an appointment basis only. We deliver from stock and we do not sell what we can't back up.

Our store in Toronto is the largest in Canada; it includes a service and repair facility of 1400 square feet, salesroom of 900 square feet, general offices of 600 square feet, and storage facilities of 1000 square feet.

We maintain a minimum of two systems up for customers to work with. Additionally, we have two bench mark dual floppy systems up and running: one a Processor Technology Hellios II based system, the other an iCOM with IBM formatting.

Our product lines include TDL, IMSAI, Processor Technology, EMA Industries, North Star, Cromemco, iCOM, Volker Craig and Cybernex terminals, Centronics printers, Sanyo monitors, and OAE paper tape readers. We are also handling used equipment.

We are the exclusive Canadian distributors for National Multiplex, Seals Electronics, and Scientific Research (the

Continued on page 105

NOW $2750. AMERICA'S LOWEST-PRICED PERSONAL COMPUTER SYSTEM WITH COLOR VECTOR GRAPHICS.



By taking advantage of the new technologies available to the industry today, we've consistently been able to give you one of the best prices on the market. Now because of great response, we can give you the best price. You can now buy the Compucolor 8001 for the reduced price of $2750. A complete stand-alone system with expanded graphics software for plotting points, vectors and bargraphs on a 160 x 192 addressable grid—in color. Eight independent background and foreground colors.

The Compucolor 8001 has an Intel 8080 CPU, 34 I/O ports and a color display with an effective band width of 75 MHZ compared to 5 MHZ for standard TV sets. In fact the Compucolor is the only totally integrated system on the market which includes a color display. You can also have special options for the Compucolor 8001 right now, including: Mini Disk Drives for extra memory, light pens and a variety of special keyboard features.

BASIC 8001 Is Easy To Learn. Compucolor's BASIC 8001 is a conversational programming language which uses English-type statements and familiar mathematical notations. It's simple to learn and easy to use, too. Especially when it comes to intricate manipulations or expressing problems more efficiently. The BASIC 8001 Interpreter runs in ROM memory and includes 26 statement types, 18 mathematical functions, 9 string functions and 7 command types for executing, loading, saving, erasing, continuing, clearing or listing the program currently in core.

Expandable Memory To 64K. The Compucolor 8001 has 11K bytes of non-destructible read-only memory which handles the CPU and CRT operating systems as well as BASIC 8001. Sockets are in place for an additional 21K of EPROM/MROM memory. The Random Access Main Memory has 8K bytes for screen refresh and scratch pad, 8K bytes for user workspace and room for 16K bytes of additional user workspace. The Compucolor also comes complete with a convenient mass storage device, Floppy Tape Memory. It's an 8-track continuous loop tape system, with a Baud rate of 4800 and an extra storage capacity of up to 1024K bytes per tape.

Color Graphics At Alphanumeric Black And White Prices.

That's what we're becoming famous for, and thanks to the tremendous response to the Compucolor 8001, we've been able to reduce our price even lower—to $2750. Look over our dealer listing on the adjacent page for the dealer nearest you. Then drop by for a demonstration. And while you're checking out the Compucolor 8001, check out your dealer's financing plan. He can help you turn a good deal into a good deal more.

Compucolor Corporation, P.O. Box 569, Norcross, Georgia 30091.

Compucolor® Corporation



The Small Computer

Twenty-five years ago a computer as powerful as the new Processor Technology Sol-20 priced out at a cool million.

Now for only $995 in kit form or $1495 fully assembled and tested you can have your own small computer with perhaps even more power. It comes in a package about the size of a typewriter. And there's nothing like it on the market today. Not from IBM, Burroughs, DEC, HP or anybody else!

It fills a new role

If you're an engineer, scientist or businessman, the Sol-20 can help you solve many or all of your design problems, help you quantify research, and handle the books too. For not much more than the price of a good calculator, you can have high level computer power.

Use it in the office, lab, plant or home

Sol-20 is a smart terminal for distributed processing. Sol-20 is a stand alone computer for data collection, handling and analysis. Sol-20 is a text editor. In fact, Sol-20 is the key element of a full fledged computer system including hardware, software and peripheral gear. It's a computer system with a keyboard, extra memory, I/O interfaces, factory backup, service notes, users group.

It's a computer you can take home after hours to play or create sophisticated games, do your personal books and taxes, and a whole host of other tasks.

Those of you who are familiar with small computers will recognize what an advance the Sol-20 is.

Sol-20 offers all these features as standard:

8080 microprocessor — 1024 character video display circuitry — control PROM memory — 1024 words of static low-power RAM — 1024 words of preprogrammed PROM — built-in cassette interface capable of controlling two recorders at 1200 bits per second — both parallel and serial standardized interface connectors — a complete power supply including ultra quiet fan — a beautiful case with solid walnut sides — software which includes a preprogrammed PROM personality module and a data cassette with BASIC-5 language plus two sophisticated computer video games — the ability to work with all S-100 bus products.

Full expansion capability

Tailor the Sol-20 system to your applications with our complete line of peripheral products. These include the video monitor, audio cassette and digital tape systems, dual floppy disc system, expansion memories, and interfaces.

Write for our new 22 page catalog. Get all the details.

Processor Technology, Box B, 6200 Hollis St., Emeryville, CA 94608. (415) 652-8080.

Processor Technology Corporation

A Catalog

of Liberating Home Computer Concepts

Ted M Lau
1219 Pioneer Dr
St Louis MO 63132

Unlike commercially developed systems, home grown computer systems are not required to justify themselves directly in terms of dollars and cents, which means that home grown systems can be put to a wide variety of uses, from whimsical games for pure enjoyment to public services for helping people, uses that are certainly worthwhile but whose worth cannot be quantified. I suspect the computer hobby will continue to address itself to a broader range of applications even after prepackaged consumer oriented systems have filled the marketplace in the next few years.

But like all powerful technologies, home computers should not be allowed to wander into any applications for which they seem suited, but should be guided by human values. Technology is often called "value free," meaning it is neither good nor bad of itself, but by virtue of the effects of its applications. To be "good," humankind (represented by each individual) must first define its will, and only then how technology can serve it.

Presented here is a catalog of home computer applications that serve the goal of "human liberation" as I personally envision it. Many of the applications listed can be found in other sources, but many well known applications are omitted. I have attempted only to include enough examples to illustrate the concepts underpinning each category so that the reader will be able to use the catalog to generate new applications of his or her own. Along the way I briefly introduce some thoughts of Marshall McLuhan, the media guru, but interested readers should consult a library for a fuller treatment.

In brief my message is: Home computers should serve your values. If you don't value war games, don't put them on your computer. I value "human liberation" and here's how I see home computers aiding that endeavor. But whether you agree with my vision or not is incidental. Use this catalog as an example. Go to it: Define your own values, and then define how the computer might help.

A. LEISURE TIME LIBERATION

1. Games

a. The computer as game board or game recorder: The game participants input their moves, the computer updates the game status display(s), and keeps move records.

   1) Go, Stratego, Risk, Avalon-Hill games (in general, games with complex boards): Using a graphics display device to represent board and token position, and a keyboard or panel for input of moves.

   2) Speed chess: Using a wired chessboard as an input device, the computer senses each player's move, keeps the clocks, and prints a list of moves at completion of the game. A move number display allows the players to add additional time periods if both players satisfy the first period requirements (rather than having the game end when one player's time expires. This is a variation that stresses speedy but not quality play).

b. The computer as kibitzer: One participant inputs all moves, and is aided in playing by advice from the computer.

   1) Chess: The computer advises of threats or potential threats against your pieces, eg: forks, skewers, checks, discoveries. Let two players, each with computerized kibitzer, compete with each other for accuracy, time, etc.

   2) Blackjack (and games of chance): The computer advises whether to stay or hit, and how much to bet. Many statistically based blackjack systems exist, some in print, waiting to be algorithmatized.[1] (Imagine the impact on Las Vegas of hordes of gambling crazed people equipped with wristwatch-sized kibitzers of this variety. Or imagine the sales potential of such a one chip blackjack calculator!)

c. The computer as opponent: The player inputs a move, and the computer responds with its own. An example is HEXPAWN by Robert Wier *[page 36, November 1975 BYTE]*, where the computer learns to play better through experience.

How about dominoes, backgammon, go-moku (five in a row), hex, football, checkers? The

computer could be made into an extremely skillful player in these games, and a difficult opponent to beat. Games that are not soluble are best.

Or get your computers together for a computer tournament, with no human players at all. The First Annual World Computer Chess Tournament was held during the summer of 1975.

2. **Simulations**
   a. The computer as an analog: The player inputs initial parameters, and optionally changes the parameters during the simulation. The computer outputs the values of variables that are described by a logical or mathematical model.

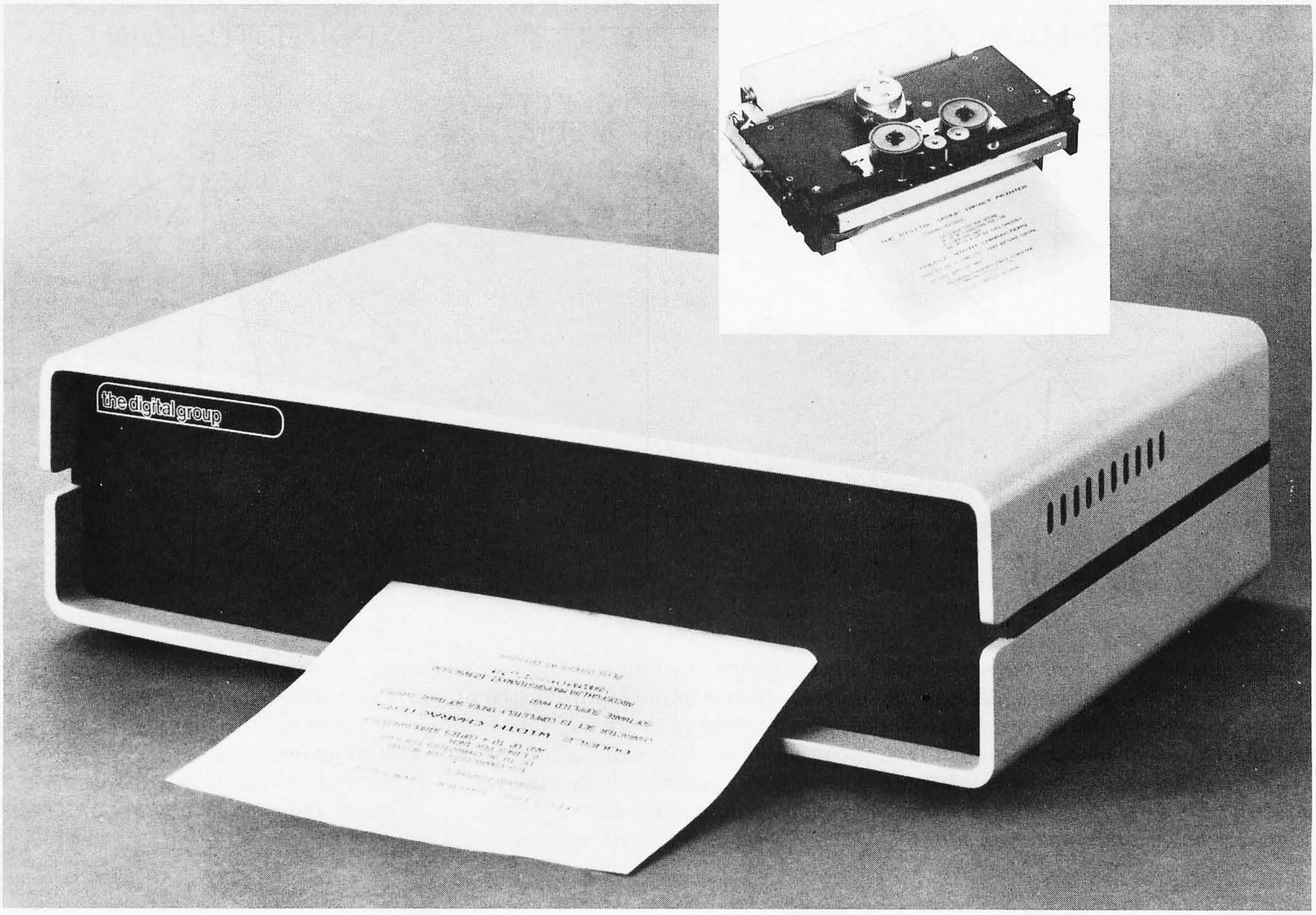
      Familiar examples are LIFE (cell colony growth), LEM (lunar landing), and ROADRACE.
      1) Patient work-up: The computer simulates a person with an illness by responding to questions posed by a medical student. The student tries to deduce the illness (make a diagnosis) by recognizing a pattern of symptoms.
      2) Environment: Human and animal population dynamics, pollutant flow up the food chain, nuclear reactor dynamics, automobile fuel economy – all can be simulated. (Simulations hackers beware, though: are you creating an accurate model or did you overlook something?[2])
3. **Toys**
   a. The computer as plaything: The infant inputs actions, and the computer responds playfully.
      1) Crib entertainer: The baby strikes crib objects (sensors), and the computer outputs funny noises, blinks lights, motivates other objects, etc.
   b. The computer as robot: The computer motivates itself around an environment, and optionally performs work.

      Examples are the maze-running mechanical rat which learns, the object-avoiding turtle, the housemaid robot, the barmaid robot, and the billiard player.
      1) Piano player (not to be confused with player piano): Robot sits on the bench, reads the notes, and plays the piano. (Or why make it humanoid at all? Simply mount controls to each key.)

Using the computer for fun avoids consuming ready-made leisure time activities. As opposed to TV ping pong, home computers are general purpose and do not come equipped with software ready for action. Computer players must first decide how to spend their leisure, and then must create the means for playing, or purchase software for the purpose. As much or more enjoyment, experience and learning occurs during the creating as during the playing.

B. MEDIA LIBERATION

1. **Translation**
   a. The computer as Seeing Eye dog: Inputs from the environment are translated by the computer into outputs in a different sensory mode. These outputs serve as substitute inputs for the user.
      1) Printed text to spoken word for the blind: A very sophisticated minicomputer based product was recently announced for this purpose. Vocal analysis, computer-generated vocalization, and text recognition are reasonable goals for advanced home computer technicians. Other translations for the blind include spatial perception to sound and spatial perception to vibration.
      2) Touch, shorthand, or sign to speech for the mute.
   b. The computer as linguistic translator: Inputs in one language are translated and output in another language.
      1) Cross assemblers: Assuming the computer hobby remains linguistically diversified, translators will be valuable tools for the promoting of free software exchange of programs in the public domain.
      2) Human language translation: Most experts have been disappointed with results of computerized translators, except when the context is specific, as in technical journals. Their experience does not eliminate the possible benefits of limited capacity English to Spanish translation machines in government offices and elementary schools in many parts of the US.
2. **Interpretation**
   a. The computer as mass media bodyguard: Semantic inputs to the computer are interpreted and output as information whose meanings differ from those of the inputs.
      1) Political "doublespeak" interpreter:[3] Takes written or spoken text and turns buzzwords into their suppressed, connotative, or implied meanings. Easy computer program, but who defines the dictionary of doublespeak terms?
      2) TV commercial replacer: The computer senses when a commercial message comes on the air and substitutes a viewer-chosen display such as a game, art graphics, trivia quiz, silence, etc.
      3) Remote lie detector: The computer measures the stress evidenced in the vocal, facial, and manneristic expressions of the person delivering a TV speech, and outputs its findings to the viewer.[4]
3. **Condensation**
   a. The computer as library card catalog: The computer stores an index to media events.
      1) Phonograph recording, cassette, magazine index: by subject, author, keyword. Computer could scan year-end index to magazine articles for input. Books and records could have covers scanned.
      2) Public library book search network: User inputs search argument and receives list of matching titles.
   b. The computer as media monitor: The computer scans many media channels and stores either entire or condensed media events whose contents match the interests input by the user. The user then retrieves the stored events.
      1) Entire verbal events: The computer stores all



Print Your Heart Out.

With help from the Digital Group, naturally.

Now, that small computer system you own or have been considering for personal or business use suddenly becomes a lot more usable—with the addition of a full-size *impact* printer from the Digital Group. A printer designed for small computers that need big output (like yours).

With the Digital Group printer, you can print your heart out...and it won't cost an arm and a leg. The Digital Group printer is available for less than $500. That's right—$500.

Just look at these specifications:

- Fast—120 characters per second
- 96 characters per line
- 12 characters per inch horizontal
- 6 lines per inch
- Makes up to 4 copies simultaneously
- Character set and pitch variable under software control—double width characters, etc.
- 5 x 7 character matrix
- Ribbon has built-in re-inkers for a life of 10,000,000 characters
- Paper can be either a standard 8½-inch roll, fanfold or cut page
- Interfaces to 8-bit parallel ports

There are lots of capabilities and outstanding features of the Digital Group printer...and (as always) the best news is our price. Kit prices start as low as $495 for the printer and interface card. It simply can't be beat.

Find out all the facts about the Digital Group printer now. Just fill in the coupon below or give us a call for the details. We think you'll find a place for our printer in your system ...and in your heart.

the digital group

P.O. Box 6528
Denver, Colorado 80206
(303) 777-7133

Quick. I want to print my heart out.
Send me all the details on your full-size impact printer.

Name ____________________

Address ____________________

City/State/Zip ____________________

Please print.

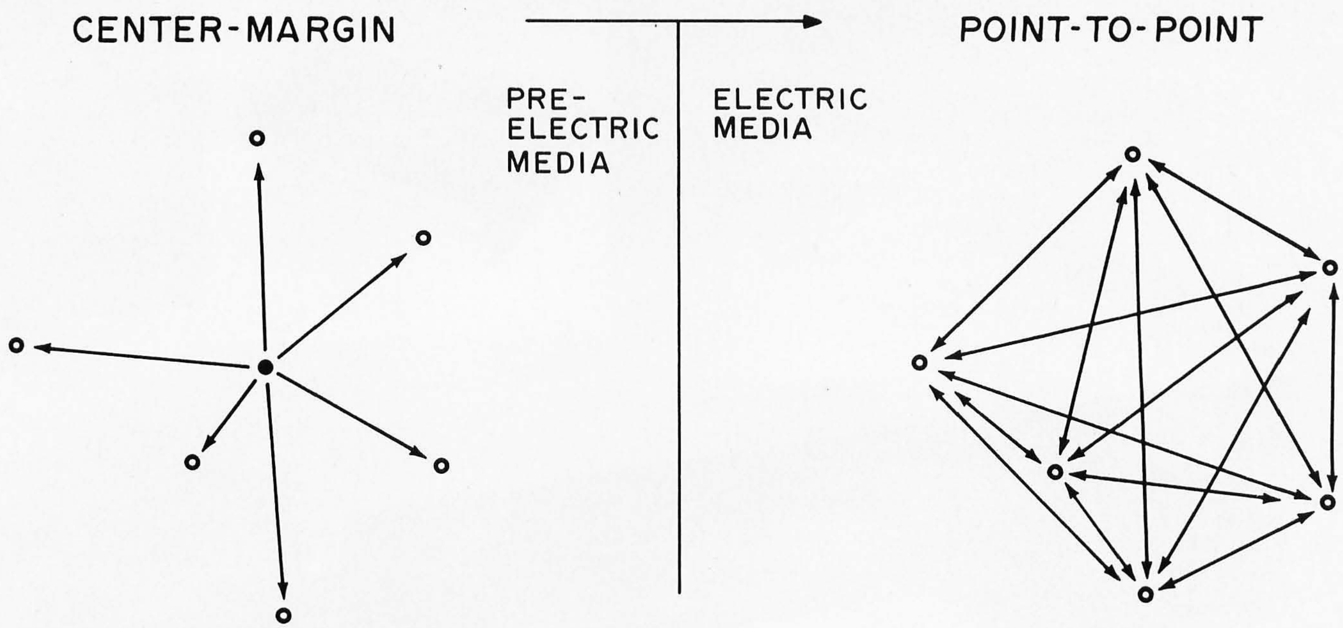


Figure 1: Communications Structures. McLuhan identifies two major classes of communications structures: center-margin, and point to point. The center margin structure at left is typical of pre-electric forms of media such as newspapers or books, in which information flows from the center to the periphery of the network. A point to point structure, as at right, typifies the many electronic forms of media such as the telephone network, ham radio, CB radio, etc. Due to political and economic factors, the communications structure of some of the electronic media such as commercial radio and television more resembles the center-margin case. But with more and more home computers, with sophisticated network terminal capabilities, the typical home communications mode will shift increasingly towards the point to point case.

TV broadcasts from all channels on video tapes, then reviews each segment for keywords, and spools off only those in which a certain level of keyword density occurs. Similar processing for AP Teletype wire, police, shortwave, and ham bands, etc.

2) Summarized verbal events: The computer monitors all radio or TV newscasts and prints a summary of nonredundant news items.

3) Music monitor: The computer monitors all stereo (or quad) FM broadcasts, identifies singers and groups by comparing to stored vocal profiles, and tapes tracks of desired artists if the event does not already exist in the library! (A challenging application perhaps, but a good way to build a personal music collection!)

4. **Alternate Media**

a. The computer as a public bulletin board: The computer stores messages from individuals to other individuals or to the public. Users input, search for, browse through, and retrieve stored messages at remote, publicly accessible terminals.

1) Carpool data base: Riders wanted, rides needed by destination and departure time are matched via a public computer network.

2) People skill resources data base: People offering skills and services match with those needing training, services, etc. A public access computer network was set up in Berkeley and was used for the above two, and many other, purposes. The network needs a scheme for categorizing the messages and message searches, as with classified ads in newspapers.

3) Interactive computerized "singles club" and dating service.

b. The computer as encoder or decoder: The computer encodes or decodes alphanumeric messages for transmission or reception over communications channel.

1) Telemetry bands: Computer-receiver combination detects and decodes scientific data in broadcasts from satellites and space probes.

2) Amateur broadcast bands: The computer interfaces between operator and transmitter and receiver for Morse CW encoding and decoding to alphanumeric, or enciphering and deciphering of private secrecy codes.

Using the computer as an interface to the mass media opens the possibility of liberating the individual from information overload caused by multichannel dissemination of largely redundant data. For example, almost all news stories on TV, radio, and in newspapers (excluding local stories) come from the two wire services, ie: are the same basic set of stories endlessly replicated. Increasing information processing power at the receiver will allow sampling of more diversified sources of news (or use of the gained free time in other ways).

Diversification also implies decentralization, since speedier information throughput (such as the personalized computer allows by enhancing both the broadcast and reception of data) on more numerous channels (say, cable TV or radio bands) will help transform our center margin

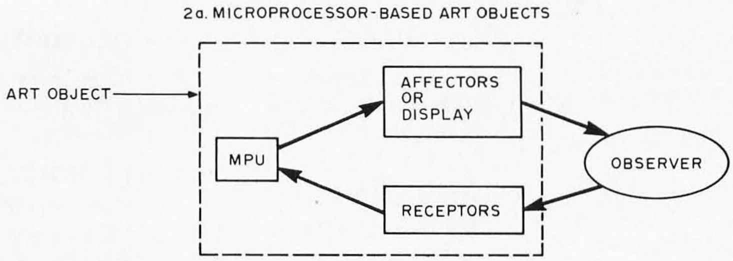


Figure 2a: A system diagram of art object components and art object to observer interaction. The interaction is identical to any computerized environment such as gaming, CAI, etc, except that the art object attempts to expose the clever operations of the computer rather than to pursue some functional purpose.

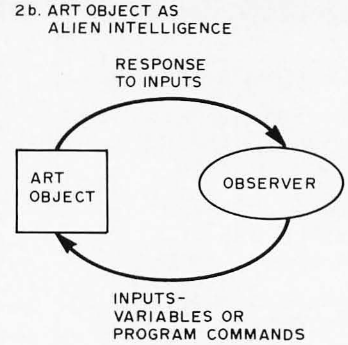


Figure 2b: The art object as an alien intelligence. In this conception, the object interprets inputs as data and program commands. Data is then transformed by the artwork and sent back out. The transformation is a complicated (but designed) function of past commands and data.

communication structure into a point to point structure (as prophesied by McLuhan[5] and represented in figure 1).

C. ARTISTIC LIBERATION

Artists will be among the first to exploit the capabilities of low cost microprocessor-based systems. John Whitney, for example, proposes to use MPUs to help create abstract animated TV graphics, a form he pioneered prior to the days of inexpensive processors.[6] Since most artists already know how they would use computers, I shall not attempt a comprehensive listing here. While I suspect that most art is liberating in some sense or other, I will restrict myself to applications that illustrate McLuhan's idea that art is an anti-environment which exposes the usually invisible effects of media on us (see figure 2a).

1. Self Exposure

a. The computer as an alien intelligence: Data inputs and program commands are input to the computer-driven artwork through means not readily apparent to the participant. The artwork transforms the data according to a program and displays. The participant attempts to deduce the transformations and program commands by varying the supposed inputs. In other words, the artwork is a black box whose inner logic the viewer attempts to deduce (see figure 2b).

1) Proximity synthesizer (theramin): The distance of the observer's body parts from hidden sensors determines pitch, filters, noise, amplitude, waveform, etc. Observer explores the object's surface. How the sensors determine the sound depends upon a Boolean concatenation of variables, eg: hand close to sensor 1 AND foot close to sensor 2 yields white noise, etc.

2) Eavesdropper: The pattern of lights displayed changes through time, but different types of changes are initiated upon detection of vocal keywords in the viewing room.

b. The computer as a Skinner box: The computer rewards certain behaviors of the viewer. The viewer may deduce that he is in a Skinner box for operant conditioning. (See figure 2c and the literature of B F Skinner.[7])

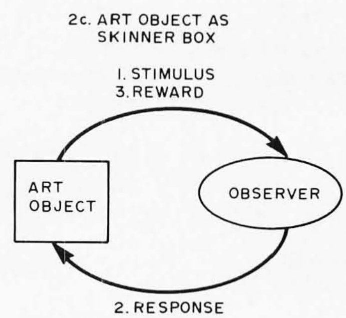


Figure 2c: Another way of viewing the art object is as a stimulus-response program. The complications and variations involved in a big program can make the object quite interesting indeed.

1) Arms above head behavioral goal: Video inputs of the viewer are displayed back to the viewer. The computer analyzes the viewer's hand positions: the higher the viewer raises his hands, the clearer his displayed self image becomes, until, overhead, the viewer can clearly see his own image. Random movements will be rewarded at first, such as raising a handbag onto the shoulder or adjusting eyeglasses.

2) Touching: If viewers touch each other, artwork dispenses a souvenir (penny, slogan button, raffle ticket).

The purpose of these computer driven artworks is to reveal to a large viewing public in graphic, intriguing, enjoyable ways, the sophistication and capabilities of computerized environments. Even for the public fortunate enough to interface with computers via CRT in museums, libraries, and schools, the computer remains transparent and hidden. These art objects can be designed to *expose* the computer's workings.

D. EDUCATIONAL LIBERATION

The problem educators and psychologists have identified as the main barrier to nationwide adoption of CAI (interactive Computer Assisted Instruction) is the high cost of hardware; but with the development of inexpensive processors, this barrier will be overcome. The benefits of CAI are well documented: More pupils master the material in less time than with conventional teaching techniques. With hardware costs decreasing, the most costly component of CAI will be software development, so if you want to do CAI, get together early with other developers on software standards so that course material can be shared. A CAI software network is a must.

Educational liberation as a computer application means every technique in this catalog. What follows is a more or less traditional listing of CAI categories, given without examples. Interested readers can pursue CAI in periodicals beginning with *Educational Technology*.

1. **Drill and Practice**
   a. The computer as drill instructor: The computer presents no new material (new material would be taught in a traditional way), but would test and reinforce learning through repetitious questioning, correcting wrong answers and accepting correct answers.
2. **Tutorial**
   a. The computer as lecturer: In "linear" fashion, the computer presents new material to the learner and periodically asks questions, but the speed or direction of presentation is not modified by the correctness or incorrectness of the answers.
   b. The computer as absent minded professor: In this "intrinsic" or "branching" mode, the computer alters its presentation depending upon the appropriateness of the previous answer only.
   c. The computer as tutor: In "adaptive" mode, the computer alters its presentation based upon the previous pattern of student responses and focuses in on weaknesses, or skips over familiar material, etc. This is the most difficult form to create.
3. **Games and Simulations**
4. **Problem Solving**
   a. The computer as a computing machine: The student uses the computer as an aid in solving a problem by inputting an arithmetic or algebraic expression, or by writing a computer program directing the computer to solve the problem.
5. **Dialogue**
   a. The computer as Socrates: The computer presents large bodies of information, and optionally queries the student or is queried by the student. The computer analyzes the student's remarks and leads the student to an understanding of his own weaknesses, need for additional study, or satisfactory comprehension, etc.
6. **Discovery Learning**
   a. The computer as a land of exploration: The computer offers a structured informational environment and the tools for its exploration. The student directs his/her own exploratory efforts.
      1) Hypertext: See Ted Nelson's *Computer Lib.*[8]

CAI promises to break down the lock step approach to learning wherein all learners, grouped by age instead of by aptitude or achievement, learn the same material at the same pace. But this promise may cause problems the experts don't foresee: CAI threatens the traditional role and authority of teachers, threatens the institutional role of public schools as keeper, babysitter and socializer of children, threatens the modern routine of family life ordered around a nine month, full time involvement by offering children the possibility of learning at home whenever they please. CAI still has a large mass of societal inertia to overcome.

E. COMMUNITY LIBERATION

"Community" can be any geographically localized group of people who have interests, problems, goals, and concerns about their lives and locality in common. One family or one block can be a community, or a city, but in general I am thinking of a neighborhood — with a traditional history, perhaps a distinct ethnic background, traditional local business and industry, topographic boundaries, and so on.

The problem communities face in our day and age is that they have been gobbled up by the cities, and have lost the wherewithal to exercise control over factors and decisions affecting their existence, issues such as housing, transportation, health care, zoning, schools, and so forth.[9] This section proposes how inexpensive computer technology might be used to benefit community development in terms of gaining two potent resources: economic power and political power.

1. **Economic Power**
   a. The computer as a service for sale: Many well known applications could be sold to customers on a per use basis, where the clients do not need full time dedicated systems of their own.
      1) Mailing list address labels.
      2) Small business accounting and billing.

      Once a local business or agency is paying for its computer with these services, it could perform additional services on a free or nominal fee basis of the sort listed in previous sections.

      3) Social service and health care referral and followup system.
      4) Part time job match service.
   b. The computer as hardware for sale: The neighborhood computer store has already arrived, with your friendly local retailer selling microprocessor hardware and related services.
      1) PROM programming services.
      2) Hardware maintenance service.
      3) Low cost peripheral manufacture.
   c. The computer as software for sale: People who are unable to travel or do heavy work (as maintenance, service, or manufacturing operations might require) could develop software, for which there will be increasing demand. System software like monitors, compilers, access methods, and sort-merge utilities; common subroutines like mathematical functions; and application packages could be developed and sold. Some corporate names I'd like to see:

Continued on page 162

Artificial Intelligence, An Evolutionary Idea

Part 1: An Overview

Michael Wimble
6026 Underwood Av
Cedar Rapids IA 52404

Artificial intelligence is a subject that has fascinated people the world over for many years, but with the introduction of the computer this fascination is becoming a reality. I do not mean the intelligence depicted in so many science fiction books and sensational movies, but rather the slow and methodical application of computers to perform tasks formerly thought to require intelligence. To attempt to even briefly describe the areas and techniques of artificial intelligence research would certainly occupy several text books. The small computer user can, however, enter this exciting field and perform personal research and experiments by using the technique described in this article.

The technique is called "Artificial Intelligence Through Simulated Evolution" and its early use and implementation is described very well by Fogel, Owens and Walsh in their book of the same name, published by John Wiley and Sons, New York, 1966. I have chosen the technique for several reasons:

1. It can be simple to implement.
2. It is a powerful technique.
3. It is readily programmable on any small general purpose system.
4. It gives the user a quick and simple introduction to the realm of artificial intelligence.

The criterion just listed represent only the low end of application of this technique. Indeed some versions of programs using this technique perform very sophisticated functions using hundreds of hours of time on large computers . . . and even involve the use of artificial intelligence to analyze and optimize the very artificial intelligence process.

The Simulated Evolution Technique What It Can Do

Before describing in detail the logic and theory of the simulated evolution technique, let us look at some early uses which are both interesting and will probably also be the initial uses of the personal computer. The technique is basically a goal directed pattern recognizer. That is, it performs some task whose success is dependent on the formation and recognition of patterns. The goal given to early programs was usually to predict some future event based upon past experience, although some other uses were in the areas of correlation analysis, image enhancement and feature extraction.

One example of an early use, which was more to debug the program than anything else, was to predict prime numbers. The computer was asked whether or not the next number was a prime. It then replied with a yes or no, and was told whether or not it had predicted correctly. The computer then remembered this historical data and looked for patterns to be used in future predictions.

To show that this technique is not just some heuristic or statistical method, let's look at a slightly different but more interesting case, that of predicting earthquakes. In the prime number example, the goal was simply to predict whether or not the next number was prime. It was given a very

simple "sense of values" that said making a good prediction was good and making a bad prediction was bad. By altering this sense of values, we change the goal.

For earthquake prediction, the goal is as follows:

- Predict when an earthquake will next occur.
- To make a good prediction is very good and to make a bad prediction is very bad.
- It is better, however, to predict a false alarm than to miss an actual earthquake.

Now let's compare the performance of the program when given the two different goals.

In the case of prime number prediction, after about the seventh observation the program realized that no multiple of two was prime. After about the seventeenth observation, the program realized that no multiple of three was prime. Unfortunately, as the numbers get larger the occurrence of primes becomes less frequent, at least for the first few hundred numbers. So the program rapidly settled into predicting that the next number will not be a prime, which is statistically very good.

In the case of earthquake prediction and using the same data as with the prime number case (ie: the occurrence of a prime number is the occurrence of an earthquake), the program acted in a much more sophisticated manner. In fact, a correct earthquake prediction was almost never missed, although 10% to 20% of its predictions were false alarms.

Finally, let's examine what is perhaps the most fascinating use of the technique: predicting people. One version of the program was set up as a game. Its goal was to predict which thing the opponent was going to do next. The opponent, a human player, was allowed to do one of two things, such as say true or false. The game was played in real time and was thus competitive. The goal of the opponent was to be unpredictable. Each time the computer correctly predicted the opponent I scored a win for the computer, otherwise it was a win for the opponent.

The opponent was free to chose any strategy as long as he was unaided. That is, he could not use random number generators or tables or other such aids. He might, for instance, choose to be as random as possible right from the start or he might try to use a particular pattern until he felt the computer had discovered it and then to use a different pattern.

One would think that if a person could make one of two choices, and he/she made choices purely randomly, then the best one could correctly predict him/her would be 50% of the time. This is true. In practice, however, an unaided human cannot perform random actions; there is always some pattern or at least a bias. Moreover, it seems that a person in a competitive situation tries to find the pattern of his/her opponent and to then develop an antipattern pattern. One proof of this, in fact, is the performance of the computer. When played against several different players and requiring at least a hundred predictions before the opponent could end the game, the computer successfully predicted the opponent 55% to 85% of the time, depending upon the opponent. Typically, 65% of its predictions were correct.

The Simulated Evolution Technique An Overview

I will now explain the general functioning of the program, first by analogy and then by actual detail. The analogy may be shocking to some, but bear in mind that it is *only* an analogy. I use it because, although it is technically inaccurate, it helps make the process clear and it provides some layman type concepts and explanations for the actual mathematical processes that occur.

As the name of the technique implies, the process involves creating an "organism" which then undergoes evolution. The organism has a survival instinct, and to survive it must successfully and continually achieve a goal. If at any time it fails, it is made to produce mutated offspring, each slightly different than the parent. These offspring along with the parent are then compared to determine which one is best able to achieve the goal, usually based upon historical data. The one best organism is then kept as a new parent and all the rest are destroyed. The new parent now survives until the next time it fails to achieve the goal.

What we then have is the simulation of thousands of years of evolution in a matter of minutes on the computer. Each organism has several basic facilities to aid in achieving its goal. It has an awareness of the goal via a matrix of values which serves both as the evolutionary selection criterion and as a sense of values or judgments. It is able to interact with the environment as in the case of predicting the environment. Finally, it has a memory. Some programs allow unlimited memory while others may somehow restrict what can be remembered. Some programs even allow the organism to choose by itself what it wishes to remember.

The importance of memory is well illus-

trated by Fogel and his co-authors (page 1):

The distinction between knowledge and intelligence became clear; knowledge being the useful information stored within the individual, and intelligence being the ability of the individual to utilize this stored information in some worthwhile (goal directed) manner.

These characteristics: goal direction, environmental interaction and memory, when coupled with one other facet, are responsible for the program's "intelligent" behavior. The single most important characteristic of the program is that it is self-organizing. It starts out with no preconceived notions. The programmer's own experiences are not reflected in the programmed organism. The program is not limited to performing only a few particular goals. What does occur is that the program is given a goal and the organism alters itself, changing its own makeup in accordance with its own judgment and experience, always becoming better able to achieve the goal. The goal can be any of a wide variety of goals, prediction being only one of the simpler goals. The goal can even change while the program is running and the organism will alter itself to meet this new goal. With this in mind, consider Fogel's offering of a definition of intelligence (page 2):

More carefully stated, intelligence can be viewed as the ability of any decision making entity to achieve a degree of success in seeking a wide variety of goals under a wide variety of environments.

There are few restrictions on the program other than those explicitly or implicitly given by the programmer. The restrictions within which the programmer must operate are as follows:

1. The amount of computer memory available determines the maximum possible sophistication of the program.
2. The programmer will usually place limitations on the evolution process so that it will stop after some time period. Thus it will not take hours or days to make the next response.
3. The organism can recognize only a fixed number of input observation types. For the game described in part 2 of this article, the number is two. The amount of computer memory required is related to the square of this number and so seldom is greater than 16.
4. The organism can make only a fixed number of output response types. For the game to be described the number is two and is the same as the input observation types (ie: the output is a prediction of the next input).
5. The program operates sequentially. It makes a response, receives feedback which it uses to judge the "goodness" of its previous response, and reorganizes itself to make the next response.

The Simulated Evolution Technique Detailed Representation

The actual program uses what is called a state space mathematical model. The model consists of a set of five elements thus:

$$\left\{ \{S\}, \{I\}, \{O\}, \{T\}, f \right\}$$

where:

$\{S\}$ is a set of states

$\{I\}$ is a set of input symbols (input alphabet)

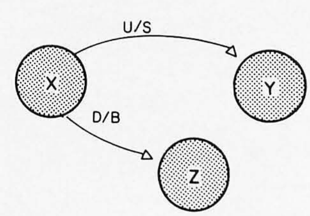
$\{O\}$ is a set of output symbols (output alphabet)

$\{T\}$ is a set of transitions

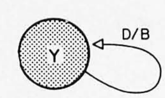
f is the current state

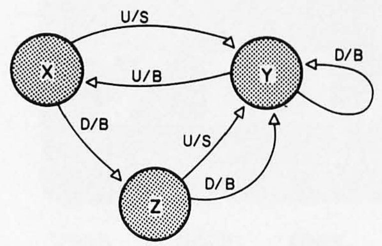
The brackets "$\{\}$" denote sets, a notion often found in mathematical texts.

As an example, figure 1 represents a hypothetical model for a stock investor. The current state, f, is state X. The model says: The model is at state X and if the market price of the stock goes up (symbol U) then the investor will sell a block of the stock (symbol S) and go to state Y; otherwise if the market goes down (symbol D) then the investor will buy a block of the stock (symbol B) and go to state Z. This is represented in figure 1 as:



Assume that the market had gone up in the previous example, the current state would now be Y and if the market price of the stock were to now go down the model says that the investor would buy a block of stock and remain at state Y. In figure 1 this is represented by:





{ S } = { X,Y,Z }
{ I } = { U,D }
{ O } = { B,S }
{ T } = { SEE TEXT }
f = X

Figure 1: Hypothetical state space model of a stock investor. The letter codes U and P stand for the up and down fluctuations of the stock market prices. The letters B and S stand for the buying and selling of stock. The set T is the set of all the transitions of the model.

For our pictorial representation of a state space model we have a set of states which is represented by the circles. The set of transitions is represented by the arrows. Alongside of each transition arrow is a pair of symbols separated by a slash. The symbol to the left of the slash is a symbol from the input alphabet and says that if the model is currently at the state connected to the tail of the arrow and this input symbol occurs, then this transition is to be taken. The symbol to the right of the slash is a symbol from the output alphabet and says that if this transition is taken, as per the previous statement, then this symbol is to be output and the new current state is the state connected to the head of the arrow.

State Space and Evolution

Given a state space model, we now look at the evolution process. There are five basic ways we can change the model; we can:

1. Add a new state (change the set $\{S\}$).
2. Remove an existing state (change the set $\{S\}$).
3. Change (repoint) a transition (change the set $\{T\}$).
4. Change the current state (change f).
5. Change an output symbol for a transition.

Changing an input symbol is not allowed. Every state must have exactly as many transitions projecting from it as there are possible input symbols. Thus when at a particular state there is a valid response for every valid input.

Note that the set of input and output symbols cannot be changed during the run of a program. Technically there are ways of allowing this change but the programming becomes highly complex, so we assume that all possible observance types and all possible desired response types are known when the program begins. *[In programming, as in physics or math, this is what is called a "simplifying assumption."]*

Also, change type five is really not something that needs to be done randomly. There is a technique called deterministic optimization that assures that the output responses are the best possible for a given model. There is, however, a desirable change which can replace change type five and will be explained at the end of the next section.

The State Space Model as an Intelligent Organism

There are some interesting properties of the state space models that get developed via this simulated evolution. The first is the tolerance to noise. Also, the model is able to repair damage to itself. Finally, if the model is knocked out of synchronism with the environment it quickly locks back into synchronism.

When the model is given the goal to find a pattern, it may be constrained by the programmer or may have decided on the basis of previous inputs that the pattern for which it is searching is a simple pattern. The environment may then contain high levels of random signals (noise) but the model will automatically ignore these random elements, deciding that their pattern is too complex. It will then lock onto the underlying patterned signal and thus remove it cleanly from the noise.

If the model is suddenly damaged, such as by loss of power in a memory module, or even by programmed destruction of a state or a transition, the model tends to "heal" itself as new data is collected. It just so happens that the model is always striving to be a perfect representation of the environment. If this representation is changed, equivalent, say, to a person losing a leg, then the evolution process will tend to restore the damage to regain that optimum representation.

A common problem that does exist from this method of artificial intelligence is the loss of synchronism with the environment. What may be an otherwise perfect model may consistently yield incorrect predictions simply because it started out at the wrong state. Fortunately, as with the other problem types, the model has a tendency to lock itself back in synchronism with the environment. Since, however, this problem can occur for so many reasons, we opt to provide an evolutionary function to speed up this lock. We then replace the old change type five with a new type, "advance the current state ahead by one state." We are

Continued on page 44

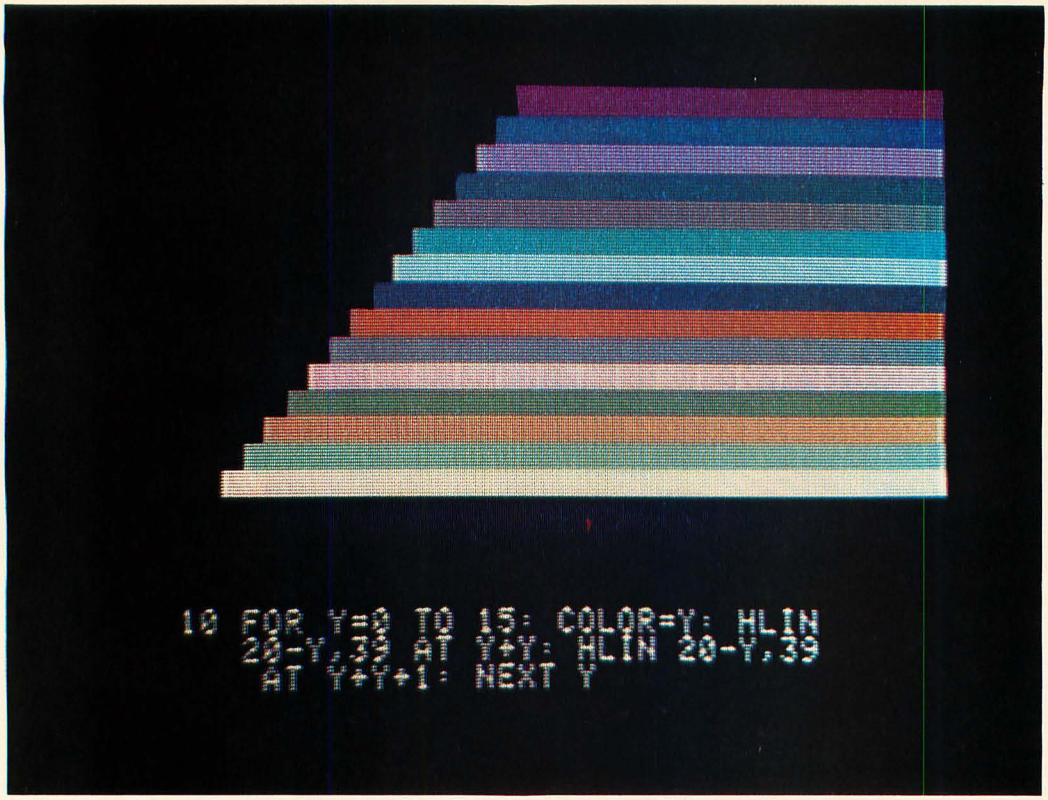


Photo 1: A color test chart showing the 15 shades of hue available from the Apple-II as presented on a typical commercial color set, using one of several RF modulators available on the market. The Apple BASIC program used to generate this color is shown in the text portion of this split screen (graphics and text) display.

System Description

The Apple-II

Stephen Wozniak
Apple Computer Co
20863 Stevens Creek Blvd B3-C
Cupertino CA 95014

To me, a personal computer should be small, reliable, convenient to use and inexpensive.

The Apple-I, my first video oriented single board computer, was designed late in 1975 and sold by word of mouth throughout California and later nationwide through retail computer stores. I think that the Apple-I computer was the first microprocessor system product on the market to completely integrate the display generation circuitry, microprocessor, memory and power supply on the same board. This meant that its owner could run the Apple BASIC interpreter with no additional electronics other than a keyboard and video monitor. The Apple-I video computer board was originally intended as a television terminal product which could also operate in a stand alone mode without much in the way of memory, although it did have a processor, space for 8 K bytes of 4 K dynamic memory chips, and its shared video generation and dynamic memory refresh logic. Apple-I was sold as a completely assembled and tested processor board with a price under $700 at the retail level.

The latest result of my design activities is the Apple-II which is the main subject of this system description article. The Apple-II builds upon this idea by providing a computer with more memory capability, a read only memory (ROM) BASIC interpreter, color video graphics as well as point graphics and character graphics, and extended systems software.

Integral Graphics

A key part of the Apple-II design is an integral video display generator which directly accesses the system's programmable

memory. Screen formatting and cursor controls are realized in my design in the form of about 200 bytes of read only memory which are built into the Apple-II's mask programmed 8 K bytes of read only memory. A 1 K byte segment of the processor's main memory is dedicated to the display generator, although it is also accessible to programs. The display transfer rate is the time it takes to fully define the contents of this segment of memory, and averages about 1000 characters per second, limited primarily by the software scrolling routines in the system read only memory.

Since the Apple-II incorporates this display generator as a part of its design, its text mode becomes the terminal for the system. The display has 24 rows of 40 characters displayed on an ordinary black and white or color television screen. Each character in the Apple-II design is a 5 by 7 dot matrix, so the present version of the system only implements upper case characters of the 6 bit ASCII subset, as well as the usual numbers

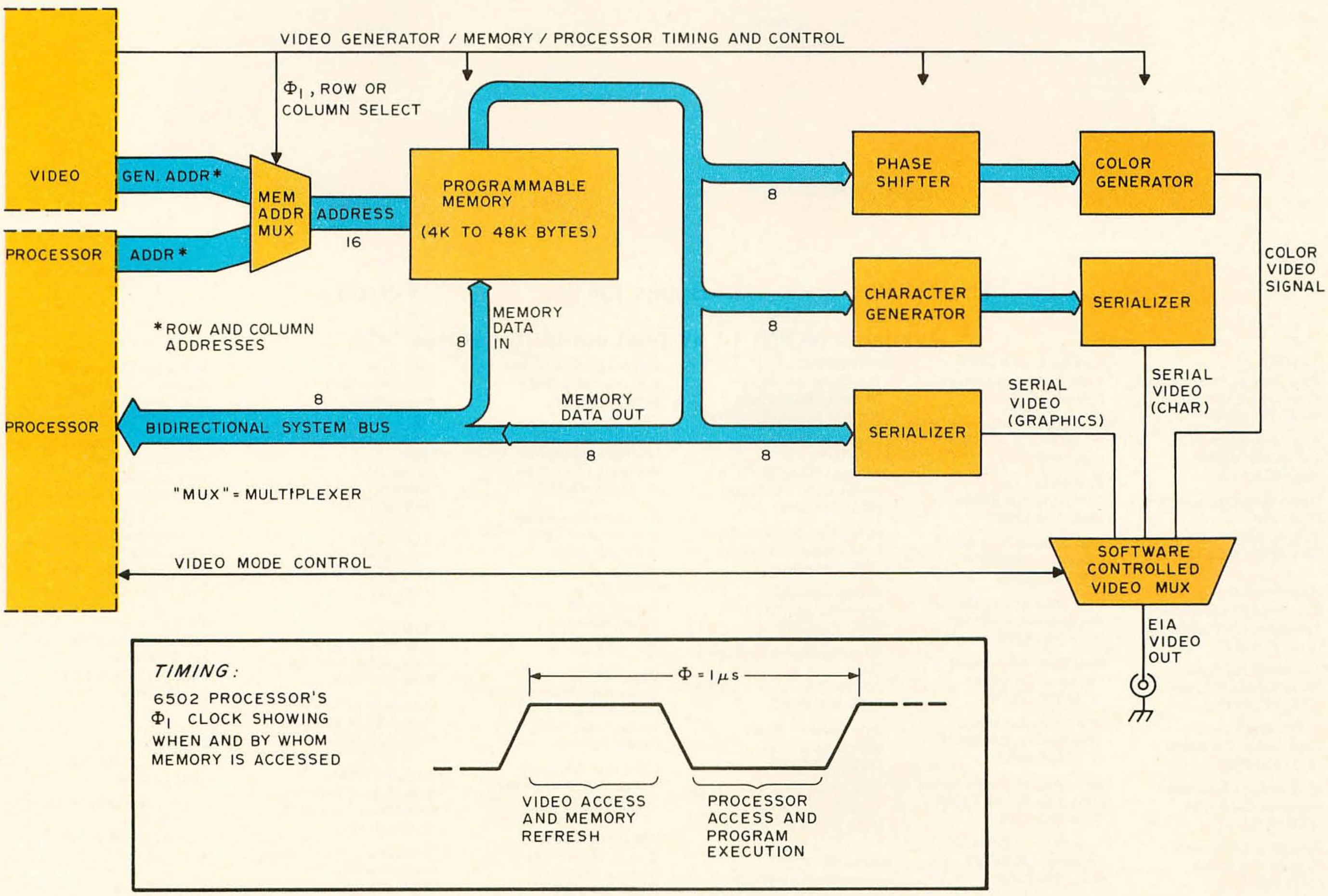


Figure 1: A block diagram of the Apple-II display generator. The generator sneaks into memory on the externally unused phase of the 6502 processor's 2 phase clock. The output of the memory is processed (after a 1 clock cycle delay) to produce a net video output through a software controlled video multiplexer. The three major modes of operation are:

Color graphics, in which each 4 bit nybble of the byte is treated as a color definition code by the color generator.

Character generation in which the 8 bit code is processed with a read only memory to generate a dot matrix pattern which is serialized and sent to the video multiplexer.

Black-white point graphics in which the 8 bit word from memory is used to control the contents of a segment of a 280 by 160 point grid.

The timing diagram shows how the basic 1 μs processor cycle period is split up into a video memory cycle and a microprocessor memory cycle. Since the processor is engaged in internal housekeeping operations during the first (high level) half of a Φ_1 period, this segment of time can be used by the video generator to sneak into memory. Since all of memory is continuously being scanned by the low order bits out of video generator, the entire 48 K byte field (maximum) of dynamic memory is refreshed by the video portion of the cycle. (Refreshing of dynamic memory means scanning through all possible low order addresses to recharge the internal memory capacitors of the chips.)

THE BASIC SOFT



The "HOW-TO-DO IT" Books for the "DO-IT" Person

*** * Available NOW! ! ! at most computer stores * ***

ARIZONA
Byte Shop of Phoenix-West
Phoenix, AZ 80529
(602) 942-7300

Byte Shop of Phoenix
Tempe, AZ 84281
(602) 894-1129

Desert Data Computer Store
POB 1334
Tucson, AZ 85702
Gary Miller & Bud Ward

CALIFORNIA
Byte Shop of Lawndale
Lawndale, CA 90260
(213) 371-2421

The Computer Store
Santa Monica, CA 90401
(213) 451-0713

A-VID Electronics
Long Beach, CA 90806
(213) 426-5526

Byte Shop of Pasadena
Pasadena, CA 91101
(213) 684-3311

Byte Shop of Tarzana
Tarzana, CA 91356
(213) 343-3919

Byte Shop/Thousand Oaks
Thousand Oaks, CA 91360
(805) 497-9595

Computer Components
Van Nuys, CA 91411
(213) 786-7411

People's Computer Shop
Sherman Oaks, CA 91423
(213) 789-7514

Upland Computer Labs
Upland, CA 91786
(714) 981-1503

Byte Shop of San Diego
San Diego, CA 92111
(714) 565-8008

Byte Shop of Westminster
Westminster, CA 92683
(714) 894-9131

The Computer Mart
Orange, CA 92667
(714) 633-1222

Byte Shop of Santa Barbara
Santa Barbara, CA 93101
(805) 966-2557

Byte Shop of San Fernando Valley
Tarzana, CA 93156
(213) 343-3919

Byte Shop of Fresno
4131 E. McKinley Ave
Fresno, CA 93703

Byte Shop of Mountain View
Mt. View, CA 94040
(415) 969-5464

Byte Shops, Inc.
Sunnyvale, CA 94086
(408) 734-9000

Byte Shop of Palo Alto
Palo Alto, CA 94306
(415) 327-8080

Byte Shop of San Mateo
San Mateo, CA 94430
(415) 341-4200

Byte Shop of Hayward
Hayward, CA 94541
(415) 537-BYTE

The Computer Shack
San Leandro, CA 94577
(415) 895-9363

Byte Shop of Diablo Valley
Walnut Creek, CA 94596
(415) 933-6252

Byte Shop of Berkeley
Berkeley, CA 94703
(415) 845-6366

Byte Shop of San Rafael
San Rafael, CA 94901
(415) 457-9311

Byte Shop of Santa Clara
Santa Clara, CA 95051
(408) 249-4221

Byte Shop of San Jose
San Jose, CA 95123
(408) 226-8383

Byte Shop of Campbell
San Jose, CA 95124
(408) 377-4685

Byte Shop of Sacramento
Citrus Heights, CA 95610
(916) 726-2557

Computerworld Stores
1309 Court Street
Redding, CA 96001
William J. Fraser

COLORADO
Byte Shop/Arapahoe Co.
Englewood, CO 80110
(303) 761-6232

The Computer Hut
Denver, CO 80202
(303) 573-4895

Byte Shop of Boulder
Boulder, CO 80301
(303) 449-6233

CONNECTICUT
The Computer Store
63 South Main Street
Windsor Locks, CT 06096

FLORIDA
Williams Radio & TV, Inc.
Jacksonville, FL 32206
(904) 354-5460

Byte Shop of Cocoa Beach
Cocoa Beach, FL 32931
(305) 784-1881

Computer Hut
Miami Lakes, FL 33014
(305) 821-2667

Micro Computer Systems & Sales
Pompano Beach, FL 33068
(305) 972-6093

Byte Shop of Miami
Miami, FL 33155
(305) 264-BYTE

Micro Computer Systems Inc.
Tampa, FL 33609
(813) 879-4301

ILLINOIS
American Microprocessor Equipment & Supply Corp.
Prairie View, IL 60069
(312) 634-0076

INDIANA
Graham Electronics
Indianapolis, IN 46204
(317) 634-8202

The Home Computer Store
Indianapolis, IN 46204
(317) 894-3319

The Data Domain
Bloomington, IN 47401
(812) 334-3607

Computer Specialists
W. Lafayette, IN 47906
(317) 743-1711

KENTUCKY
The Data Domain
Lexington, KY 40502
(606) 233-3346

MARYLAND
Computer Workshop
Rockville, MD 20852
(301) 468-0455

Science Education Ext. Corp.
Potomac, MD 20854
(301) 299-9506

MASSACHUSETTS
Computer Mart, Inc.
Waltham, MA 02154
(617) 899-4540

Computer Warehouse
Boston, MA 02215
(617) 261-2701

MINNESOTA
Byte Shop of Eagan
Eagan, MN 55121
(612) 452-1841

MISSOURI
Computer Systems Center of St. Louis Inc.
Chesterfield, MO 63071
(314) 576-5020

Computer Workshop
Kansas City, MO 64152
(816) 741-5055

NEW HAMPSHIRE
Computer Mart of NH
Nashua, NH 03060
(603) 883-2386

Microcomputers Inc.
Nashua, NH 03060
(603)889-1646

NEW JERSEY
Hoboken Computer Works
Hoboken, NJ 07030
(201) 420-1644

The Computer Mart of New Jersey
501 Route 27
Iselin, NJ 08830

NEW YORK
The Computer Mart of New York
New York, NY 10001
(212) 279-1048

Synchro-Sound Enterprises
Hollis, NY 11423
(212) 468-7067

Computer Mart of Long Island
East Meadow, NY 11554
(516) 79400
(516) 794-0510

Byte Shop of Levittown
Levittown, NY 11756
(516) 731-8116

New York
Mini Micro Mart
Syracuse, NY 13203
(315) 422-4467

OHIO
Digital Design
Cincinnati, OH 45243
(513) 561-6733

OKLAHOMA
High Technology
Oklahoma City, OK 73116
(405) 842-2021

OREGON
Byte Shop of Beaverton
Beaverton, OR 97005
(503) 644-2686

Byte Shop of Portland
Portland, OR 97201
(503) 223-3496

PENNSYLVANIA
Byte Shop of Bryn Mawr
Bryn Mawr, PA 19010
(215)525-7712

Personal Computer Corp.
Frazer, PA 19355
(215) 647-8460

RHODE ISLAND
Computer Power Inc.
Warwick, RI 02886
(401) 738-4477

SOUTH CAROLINA
Byte Shop of Columbia
Columbia, SC 29205
(803) 771-7824

TENNESSEE
Byte Tronics
Nashville, TN 37203
(615)329-1979

Micro Computer Systems
Knoxville, TN 37922
(615) 966-9849

TEXAS
Micro Store
Richardson, TX 75080
(214) 231-1096

K.A. Electronics
Dallas, TX 75247
(214) 634-7870

Electrotex
Houston, TX 77006
(713) 526-3456

Altair Computer Center
Houston, TX 77036
(713) 780-8981

Interactive Computers
Houston, TX 77036
(713) 781-2703

The Computer Shop
Vanguard Systems
6812 San Pedro
San Antonio, TX 78216

Southwest Technical Products Corp.
POB 32040
San Antonio, TX 78216
Gary Kay

UTAH
Byte Shop/Salt Lake City
216 S. State St.
Salt Lake City, UT 84111
Bob Bollinder

VIRGINIA
Shire Enterprises
Richmond, VA 23222
(804) 321-4560

Virginia Computer Hobbies Unlimited
Richmond, VA 23235
(804) 276-5056

WASHINGTON
Almac/Strom Electronics
Seattle, WA 98108
(206) 763-2300

WISCONSIN
The Milwaukee Computer Store
Milwaukee, WI 53213
(414) 259-9140)

Microcomp
Fond Du Lac, WI 54935
(414) 922-2515

CANADA
Trintronics
Toronto, Ontario
Canada M5V 1Z1
(416) 598-0262

Byte Shop of Vancouver
Vancouver 9 B.C.
(604) 736-7221

The Pacific Computer Store
Vancouver, B.C. V5R 2J4
(604) 438-Data

JAPAN
Byte Shop of Tokyo
2-9-9 Stotdanda
Chiyodaka, Tokyo
Kiyotake Ikeda

WARE LIBRARY

This LIBRARY is a complete do it yourself kit. Knowledge of programming not required. EASY to read and USE

Written in compatible BASIC immediately executable in ANY computer with at least 4K, NO other peripherals needed.

This Library is the most comprehensive work of its kind to date. There are other software books on the market but they are dedicated to computer games. The intention of this work is to allow the average individual the capability to easily perform useful and productive tasks with a computer. All of the programs contained within this Library have been thoroughly tested and executed on several systems. Included with each program is a description of the program, a list of potential users, instructions for execution and possible limitations that may arise when running it on various systems. Listed in the limitation section is the amount of memory that is required to store and execute the program.

Each program's source code is listed in full detail. These source code listings are not reduced in size but are shown full size for increased readability. Almost every program is self instructing and prompts the user with all required running data. Immediately following the source code listing for most of the programs is a sample executed run of the program.

The entire Library is 1100 pages long, chocked full of program source code, instructions, conversions, memory requirements, examples and much more. ALL are written in compatible BASIC executable in 4K MITS, SPHERE, IMS, SWTPC, PDP, etc. BASIC compilers available for 8080 & 6800 under $10 elsewhere.

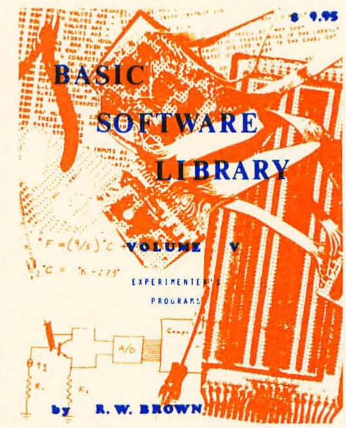
This Library is destined to become one of the reference bibles for the small computer field, due to its versatility and uniqueness and the ease of operation of the programs it contains. These volumes are deductible as a business expense when purchased by a company. Send your remittance for prompt delivery, while supplies last. Volume discounts are available to qualified dealers.

VOLUME I
Bookkeeping
Games
Pictures $24.95

VOLUME II
Math & Engineering
Plotting & Stat
Basic Statement Def $24.95

VOLUME III
Advanced Business
Billing
Inventory
Payroll $39.95

VOLUME IV
General Purpose $9.95



VOLUME V
Experimenter $9.95

FUTURE ADDITION TO THE "BASIC SOFTWARE LIBRARY"

Volume VI (A Complete Business System - $49.95) General Ledger System - Taxes, Pyrl, W-2's, Inventory, Depr., Financial Statements, etc. AVAILABLE MID SUMMER

Add $1.50 per volume for postage and handling.

SRI

SCIENTIFIC RESEARCH

1712-B FARMINGTON COURT
CROFTON MD 21114
Phone Orders call (800) 638-9194
Information and Maryland Residents Call (301)-721-1148

Photo 2: *This series of photos shows the steps in writing an animated BASIC game using the Apple-II computer's BASIC interpreter. This sequence highlights the process of writing a paddle versus "wall" game where the object of play is to knock bricks out of the wall and eventually get the ball to go all the way through. This game is similar to many seen in amusement parks and arcades, and is typical of the kind of game which can be implemented with Apple-II's BASIC software. Using the split screen graphics and text display mode, the BASIC statements are shown at the bottom of each picture.*



Photo 2a: The first step in any game is to generate the uniform color background for the action of the game. Here we use a blue field.

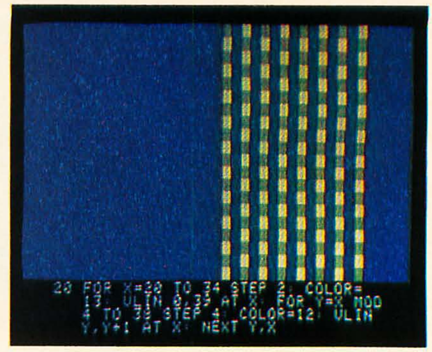


Photo 2b: Then we must add a liberal dose of obstacles and field pieces to make the problem interesting. For this game, the major obstacle is a brick wall of orangish (color 13) and greenish (color 12) bricks. Later on, since we can look at the contents of the screen directly, the game algorithm will be manipulating these bricks.

and graphics available in standard character generator read only memory parts. Assuming that the video display is the currently assigned system output device, the display is accessed through our system software in read only memory by using a subroutine called COUT which adds text to the screen using an automatic scrolling technique. This is typical of the many read only memory routines which I've incorporated into the ROM to provide complex features with relatively simple user interfaces. Another example of such a software feature is a user definable scrolling window. This means that the user of the system can pick any of four coordinates defining any rectangular subset of the viewing area of the video screen as the current scrolling zone. The remainder of the display will remain frozen and data in the window will scroll normally when COUT is accessed. This is a most useful feature: For example, the user can set up a game background or instruction menu in one part of the screen while using the remainder of the screen for scrolling the variable data.

In the text mode, each character position may be displayed in normal (white character on black background) or inverse, or flashing modes. This information is specified by the high order bits of each character stored in the display memory. The cursor position, for example, is indicated by forcing the character at the cursor location to be in the flashing mode with inverse video.

User application programs may switch the display mode from character to color graphics with a single instruction, dividing the screen instantly into a patchwork of controllable color on a grid of 40 horizontal locations by 48 vertical locations. Each cell in the grid may be one of 15 colors, and software built into the system read only memory can be used to define the color of any point as set by X and Y coordinate integer values. Photo 1 shows a color scale for the 15 colors possible, and a simple BASIC program which generated the display. Here the scrolling window features are used to set the color graphics mode in the fixed portion of the screen (above) and set the text mode of operation in the scrolling portion (below). This mixed mode provides a 40 by 40 color graphics grid plus four lines of scrolling text at the bottom of the screen. A routine in the system read only memory selects this mode and sets up the scrolling window corresponding to the text portion. I've found this mode especially useful to BASIC programmers who can write animation games like Pong while holding a traditional BASIC conversation in the text region of the screen. This split screen mode of viewing is used for all the color graphics of photo 2 as well.

The same display memory region that is used for the text display is used for the color graphics. System software routines supplied in the read only memory of the processor allow users to simply clear the display, select colors, plot points, draw horizontal and vertical lines, and sense the color values presently at specified screen positions. I like to think of these system software subroutines as enhancements to the 6502 instruction set for the purposes of display control.

High resolution graphics is the remaining Apple-II display mode. This mode of display is set up by system software routines which are delivered with the computer, but are not built into the system read only memory. (Even with 8 K bytes for the read only memory space, there sometimes isn't enough

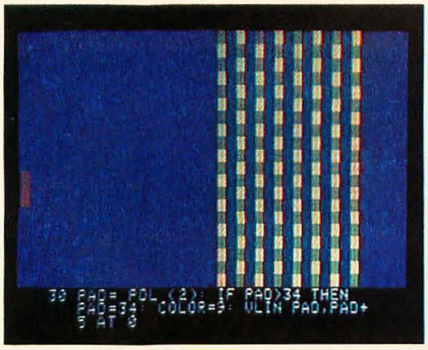


Photo 2c: Next, we must of course add a paddle, here created with a deeper yellowish orange (color 9) hue.

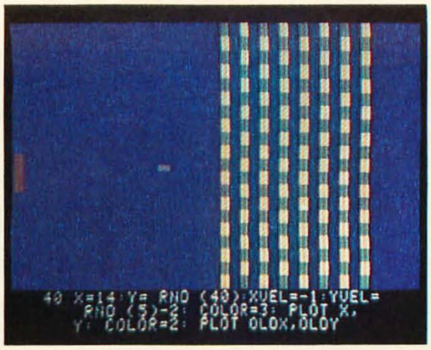


Photo 2d: Then, since no video court game is complete without a ball we must add a square "ball" to the program, and set up some of the parameters of its motion.

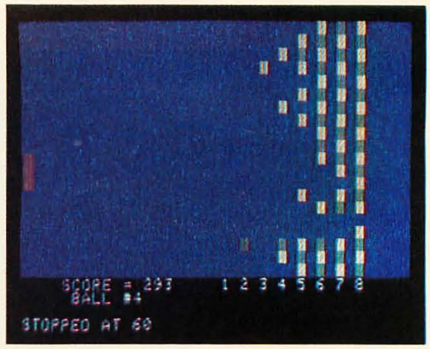


Photo 2e: Finally, the last steps in finesse are the score displays and related captions which complete the game. This game is controlled by using one of the analog inputs of the Apple-II to determine the index of the current location of the paddle, so that by twisting the pot the paddle is moved; the speaker output is used to generate a sound burst when the ball hits the paddle or wall.

room to fit all the needed features.) In the high resolution mode, 8 K bytes of main memory store the data for a display of 280 horizontal dot positions by 192 vertical dot positions; so to allow enough room for some BASIC software to play games with this mode the system requires at least 12 K of memory. If a color television is used with this high resolution mode, the available colors are black, white, violet and green. A mixed mode with 160 rows of 280 dots plus four lines of scrolling text can also be set up. Applications of the high resolution graphics modes include game boards, mazes, maps, plots and histograms, user definable character sets, and games like Space War in its original animation graphics versions.

Some Details

All the Apple-II video modes work identically, using a common clock timing chain which is shared by the processor, memory refresh and video generation logic. During each microprocessor clock cycle's Φ_1 clock pulse, an address is specified by the video circuits and directed to the programmable memory of the system through the address multiplexor (MUX) of figure 1. Display data is received by the three forms of video data generators toward the end of the Φ_2 pulse, and this data is then latched for use during the entire next clock cycle. Since all this action occurs during the Φ_1 pulse which lasts 500 ns, the video generator is able to take over the access to the memory at a time when the 6502 processor is busy with internal housekeeping and processing operations which leave the data bus free. During the Φ_2 pulse, when the processor takes command of the bus, the programmable memory of the system is used by the executing program as if the video generator didn't exist at all. Because the integrated display design uses this direct memory access technique without stealing processor cycles, it is possible to program accurate and predictable timing loops in software as if no DMA were present in the system.

Memory

It is alleged in the Santa Clara (Silicon) Valley that the microprocessor was invented to sell programmable and read only memory chips. It certainly has been the case that one microprocessor in the past would often support hundreds of memory chips, but times change. Technology has since bestowed upon us the 4 K bit and 16 K bit dynamic programmable memory chips.

Apple-II was designed to operate with the 16 pin dynamic programmable memory parts, which come in 4 K and 16 K versions which are (with some subtleties) pin for pin compatible.

The Apple-II board is supplied with sockets for three blocks of memory, each of which may be configured to use either 4 K or 16 K dynamic programmable memory parts, with intermixing allowed. This means that if you were to purchase an Apple with 4 K bytes of memory and later want to add 16 K bytes, there is no need to scrap the 4 K chips.

Dynamic memories have one design characteristic which is not present in the simpler (but more expensive) static memories. This is the fact that they use capacitive storage elements built into the chips which must be periodically recharged ("refreshed") to prevent the information from disappearing.

One of the elegant simplifications provided by a system such as the Apple-II with its built-in display is the fact that refreshing the entire memory address

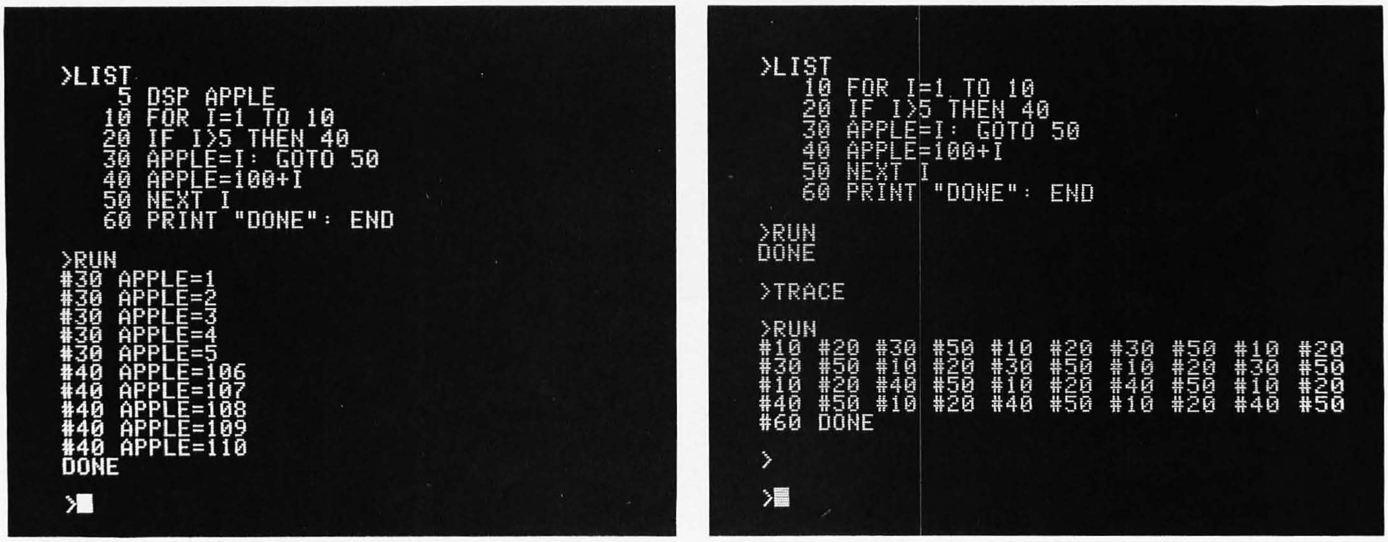


Photo 3: Two examples of the Apple BASIC interpreter, in the form of programs with several lines of execution results. (a) The interpreter has a symbolic trace feature which allows dumping of named variables whenever a change occurs. This simple program illustrates this "DSP" command with a simple computational program. (b) A similar debugging feature of Steve Wozniak's Apple BASIC interpreter is a method of running the interpreter with a statement number trace, by giving a TRACE command instead of RUN in the command mode of the interpreter. This enables one to fairly quickly debug a BASIC program by examining its effect on variables or its course of evolution through statement numbers.

space of dynamic memory chips is inherent in the operation of the video display generator. On successive pulses of the video display, it cycles through all the low order addresses of the memories as the memory is scanned to generate the video image. But scanning through the addresses within the maximum allowable time is the algorithm used to accomplish the required refreshing of the memories; so with this video generator integral to the computer, refreshing of the memories happens to come for free and is totally transparent to the user with no extended, missing or delayed cycles. This characteristic is sometimes called "hidden refresh."

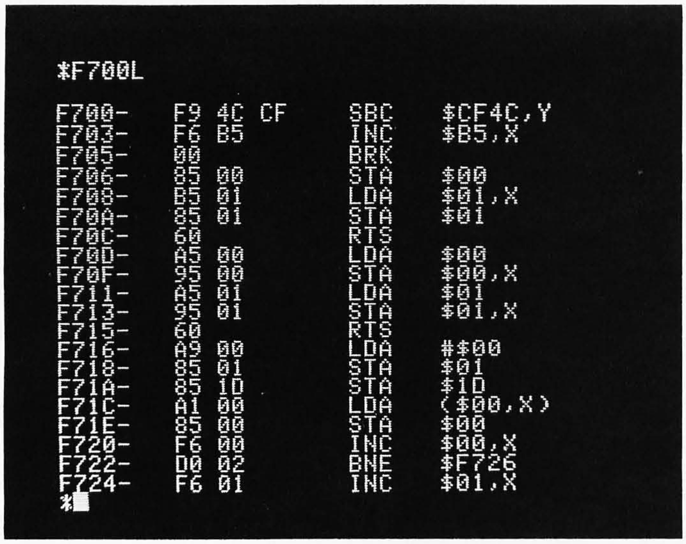
Standard Peripherals

I designed the Apple-II to come with a set of standard peripherals, in order to fit my concept of a personal computer. In addition to the video display, color graphics and high resolution graphics, this design includes a keyboard interface, audio cassette interface, four analog game paddle inputs (for user supplied potentiometers which vary a resistance which the processor measures), three switch inputs, four 1 bit annunciator outputs, and even an audio output to a speaker. Also part of the Apple-II design is an 8 slot motherboard for IO which has a fully buffered bus, prioritized interrupts, two prioritized direct memory access (DMA) schemes, and address decoding at the individual slots so that multiple bit address decoders are not required on peripheral boards.

The Apple-II cassette interface is simple, fast, and I think most reliable. The data transfer rate averages over 180 bytes per second, and the recording scheme is compatible with the interface used with the Apple-I. This tape recording method can be used with any inexpensive recorder, but as with any such use of audio media only high quality tapes should be used in order to avoid problems due to dropouts from poor oxide coatings on the tapes. In the Apple audio cassette interface, timing is performed by software which is referenced to the system clock. A zero bit is defined as a full cycle of a 2000 Hz signal (500 μs long), while a one bit is defined as a full cycle of a 1000 Hz signal (1 ms long). While reading data, full cycles are sampled, never half cycles, a method which tends to provide immunity to DC offset and other forms of distortion. All the cassette management routines are available to user programs as subroutine calls from assembly language directly, or through hooks in the BASIC interpreter.

The Apple-II analog game control paddle circuits are based upon inexpensive timer chips of the 555 type. I've used a quad timer of this type, called the 553, as shown in figure 2. To read the value of resistance on the paddle's potentiometer, the timer is strobed under software control using routines in the system read only memory. The

(a)



(b)

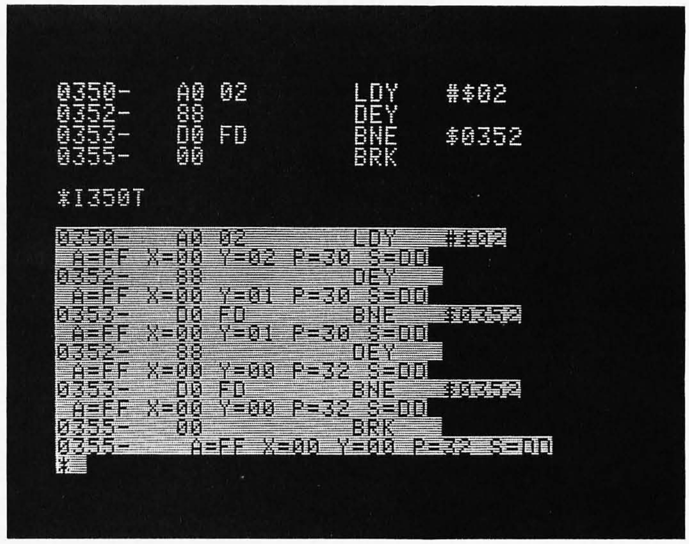


Photo 4: Far from being limited to interpretive integer BASIC, the Apple-II includes some powerful debugging and software development aids at the machine language level. Here at (a) is an example of its dissassembler mode of operation, invoked by the L command following an address in hexadecimal. A corresponding nonsymbolic assembler program will perform transformations in the other direction from text sources. Here at (b) is an example of the instruction trace command, which allows a machine language program to be followed mnemonically via dynamic disassembly, with register and condition code contents indicated after each instruction.

input routine then enters a loop which counts the length of the timer output pulse, which is a function of the paddle potentiometer's setting. To prevent endless loops if a wire breaks, the paddle scan routines exit at the maximum count of 255. The resolution of the loop is 12 μs per count.

One memory address is dedicated to the audio output port which drives a speaker. When this memory location is referenced from a program, with either a read or a write operation, the speaker drive line is toggled. Generating tones requires continuous speaker toggling by this method, at an audible rate. The cassette output port works in a similar (toggle) fashion to generate audio tones for the tape. The annunciator outputs each have two corresponding addresses, with one used to set the output and the second used to clear the outputs. Switch, paddle and cassette inputs place their data on the system bus in the sign bit position when their corresponding addresses are referenced; this choice of wiring enables software to test the state of the bit directly with a conditional branch instruction of the 6502 processor.

Apple BASIC

Apple-II comes with an Apple BASIC interpreter in the mask programmed read only memories of the system. There is no need to load it off tape, nor to dedicate any programmable memory for it. It's always there and it is impossible to accidentally clobber it. This BASIC is essentially similar to any BASIC with the exceptions that it only implements 16 bit fixed point arithmetic. It also features some unique language extensions to take advantage of the Apple-II hardware features such as color graphics and to provide conveniences in the form of debugging aids. It is intended primarily for games and educational uses.

A monitor command puts you into BASIC mode, which is indicated on the screen by a prompt character, ">". Memory limits for BASIC source programs and data are set automatically at the time of entry, but these limits may be varied by user commands. While in BASIC mode, statements are entered on the current system input device, which is normally the keyboard.

Apple-II BASIC is implemented as a translator-interpreter combination. When a line is read from the input device, the translator analyzes it and generates a more efficient internal language facsimile. Syntax errors are detected at this time. The "nouns" of this internal language are variable names, integer constants (preconverted to binary for execution speed enhancement), and string constants. The "verbs" are 1 byte tokens substituted for keywords, operators and delimiters. Because the translator distinguishes syntax, different verbs are as-

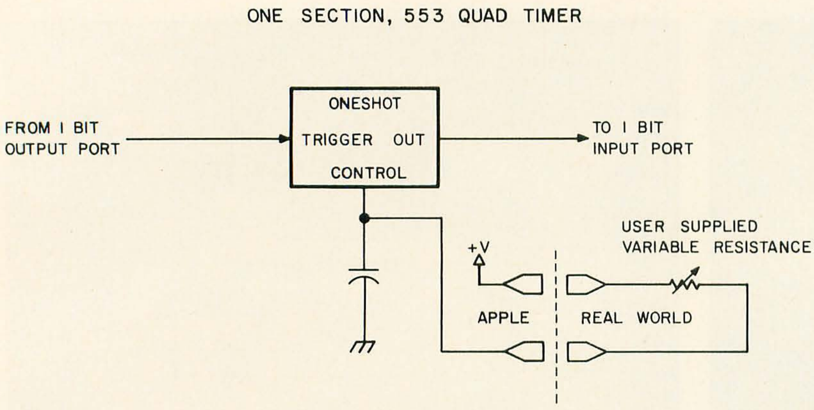


Figure 2: How to make a 1 bit measurement of an analog parameter for games (or perhaps we should say "2 bit"). Basically, a 555 style timing element is set up so that it can be triggered by a 1 bit output port. After triggering the oneshot, the processor enters a timing loop continuously testing the 1 bit input port until the end of the oneshot's cycle, which is controlled by the game parameter potentiometer. The result is an integer count developed by the timing loop which gives a measure of how long the oneshot pulse lasted, and hence a measure of the position of the input potentiometer. Apple-II implements four of these resistance measuring ports (which have plenty of accuracy for game contexts with graphics display feedback but are hardly not to be interpreted as having any absolute accuracy independent of hand-eye coordination).

signed to different usages of the same symbol. For example, three distinct verbs represent the word PRINT, depending on whether it is immediately followed by a string source, an arithmetic expression or nothing. Thus this distinction need not be made at execution time. For each verb there exists a subroutine to perform that specific action. Listing a program actually involves decompiling the internal language back to BASIC source code. Those statements with line numbers are stored as part of the user program, while those without line numbers are executed immediately. If desired, the Apple BASIC interpreter's editing functions can be set to generate line numbers automatically. Although some commands are valid only for immediate execution and others only for programmed execution, most can be employed in both ways. In the BASIC source programs, multiple statements may reside on the same line, separated by colons (':').

BASIC language statements are stored in user memory as they are accepted and variables are allocated space the first time they are encountered during immediate or programmed execution. When a program terminates, whether by completion, interruption or error conditions, all variables are preserved. Programs may be interrupted in execution by typing an ASCII control C; it is then possible to examine and modify a few variables in immediate mode, then continue execution at the point of interruption by typing the CONtinue command. BASIC provides the line number of the statement as the point of interruption when this sequence is used. The entire variable space is cleared to zero when BASIC is initialized by the CLR command, and prior to executing the RUN command. (It is possible to carry variables from one program to another, but to initiate the second program a GOTO command must be used instead of RUN in order to override the automatic clear at the beginning of execution of a new program.)

The interpreter consists of a standard expression evaluator and a symbol table routine for allocating variable storage similar to those described by Prof Maurer in his 2 part series in the February and March 1976 issues of BYTE. As statements are scanned, nouns and verbs are encountered. Variable names result in calls to the symbol table routine which pushes address and length information on the noun stack (operand stack). Constants are pushed directly onto this stack. Verbs are pushed onto the verb stack (operator stack) after popping and executing any verbs of greater priority. A verb is executed by calling its associated subroutine. Tables define priorities and routine entry addresses for all verbs. Keywords such as THEN or STEP, and delimiters such as commas and parentheses, are dealt with just as though they were arithmetic operators. Verb routines obtain their arguments from the noun stack. Because verbs such as parentheses tend sometimes to be of low, and other times of high priority, each verb is actually assigned two priorities (left hand-right hand). One represents its tendency to force execution of other verbs, the second its tendency to be executed.

Interactive Monitor

The entry into BASIC, as well as other user oriented features of the Apple-II, is provided by an interactive keyboard monitor which serves as an aid to writing and debugging machine language programs for the 6502 processor of the system. The user enters commands from the keyboard specifying data and address parameters in hexadecimal. Multiple commands are permitted on the same line and editing features facilitate error correction. I completely wrote and debugged Apple BASIC using the monitor as my only software development tool. It was of course the first hand assembled program I wrote for the system. In addition to the direct monitor commands, a number of subroutines were included in the Apple-II's mask programmed system read only memory

Author's Note

So as not to slight their efforts, I would like to thank Allen Baum for originating the Apple-II debug software, Doug Kraul for helpful suggestions on the IO structure, and Randy Wigginton and Chris Espinosa for many long and late hours testing the Apple BASIC. . . . SW

Sweet Sixteen Calling Sequence:

```
20 89 F6       -- -- -- -- -- -- --   |   -- -- --
\________/     \___________________/  |   \______/
   JSR            SWEET 16                  6502
 SWEET16          OP CODES       SWEET16    CODE
                                 RETURN
 (leave 6502                     OP CODE
direct execution)                (reenter direct
                                 6502 execution)
```

SWEET16 OP CODES (16 Bit Operands, 2's Complement Arithmetic)

Op Code	Instr Length	Description	Op Code	Length	Description
00	1	Return to 6502 mode	—	—	—————————
01	2	Branch always	1R	3	R←2 byte constant (Load register immediate)
02	2	Branch no carry	2R	1	ACC←R
03	2	Branch on carry	3R	1	ACC→R
04	2	Branch on positive	4R	1	ACC←@R, R←R+1
05	2	Branch on negative	5R	1	ACC→@R, R←R+1
06	2	Branch if equal	6R	1	ACC←@R double
07	2	Branch not equal	7R	1	ACC→@R double
08	2	Branch on negative 1	8R	1	R←R−1, ACC←@R (pop)
09	2	Branch not negative 1	9R	1	R←R−1, ACC→@R
0A	1	Break to monitor	AR	1	ACC←@R (pop) double
0B	1	No operation	BR	1	COMPARE ACC to R
0C	1	No operation	CR	1	ACC←ACC+R
0D	1	No operation	DR	1	ACC←ACC−R
0E	1	No operation	ER	1	R←R+1
0F	1	No operation	FR	1	R←R−1

Notes.

1. All branches are followed by a 1 byte relative displacement. Works identically to 6502 branches.
2. Only ADD, SUB and COMPARE can set carry.
3. Notation:
   R = a 16 bit "register" operand designation, one of 16 labelled 0 to 15 (decimal), 0 to F (hexadecimal).
   ACC = register operand R0.
   @R = indirect reference, using the register R as the pointer.
   → ← } = assignment of values.
4. Length of instructions:
   Branches are always two bytes: op code followed by relative displacement.
   Load register immediate (1R) is three bytes: the hexadecimal op code 10 to 1F followed by the 2 byte literal value of a 16 bit number.
   All other instructions are one byte in length.

The Apple-II monitor read only memory also contains an interpreter program called SWEET16 which can be used from machine language programs to implement 16 bit arithmetic operations. This facility can prove quite useful, for example, in calculating addresses, and serves as an extension of the instruction set of the 6502 which is reached by the JSR SWEET16 escape sequence in code.

to provide easy access to hardware features. These are the service routines which are used by the monitor, as well as BASIC and any user routines you care to code.

The Story of Sweet Sixteen

While writing Apple BASIC, I ran into the problem of manipulating the 16 bit pointer data and its arithmetic in an 8 bit machine.

My solution to this problem of handling 16 bit data, notably pointers, with an 8 bit microprocessor was to implement a non-existent 16 bit processor in software, interpreter fashion, which I refer to as SWEET16.

SWEET16 contains sixteen internal 16 bit registers, actually the first 32 bytes in main memory, labelled R0 through R15. R0 is defined as the accumulator, R15 as the program counter, and R14 as a status register. R13 stores the result of all COMPARE operations for branch testing. The user accesses SWEET16 with a subroutine call to hexadecimal address F689. Bytes stored after the subroutine call are thereafter interpreted and executed by SWEET16. One of SWEET16's commands returns the user back to 6502 mode, even restoring the original register contents.

Implemented in only 300 bytes of code, SWEET16 has a very simple instruction set tailored to operations such as memory moves and stack manipulation. Most op codes are only one byte long, but since she runs approximately ten times slower than equivalent 6502 code, SWEET16 should be employed only when code is at a premium or execution speed is not. As an example of her usefulness, I have estimated that about 1 K bytes could be weeded out of my 5 K byte Apple-II BASIC interpreter with no observable performance degradation by selectively applying SWEET16.■

Continued from page 32

Current State	Input Observation A	B	C	...
R	output: response A new current state: R	output: response B new current state: R	output: response C new current state: R	
S	output: response A new current state: S	output: response B new current state: S	output: response C new current state: S	
T	output: response A new current state: T	output: response B new current state: T	output: response C new current state: T	
⋮				

current state = S

Figure 2: State space model in matrix representation. The model can be described by an n by m matrix. The number of states in the model is n and the number of input types is m. Each element describes the output response and the new current state to which the next move is made.

now complete in the definition of the mutation types desired in our evolution simulation.

The State Space Model and Its Computer Representation

We now have sufficient knowledge to begin examining the computer implementation of this artificial intelligence by simulated evolution. Although the discussion will tend to be general, the emphasis is on using the technique as a two symbol pattern recognition game. This game would then allow the computer programmer to experiment on his/her own system with predicting primes, earthquakes, people or any of the virtually limitless areas of pattern recognition and feature extraction. Those programmers with a more extensive background in mathematics and automata theory and with larger computer facilities at their disposal may want to attempt more sophisticated programs with many symbol pattern recognition, advanced evolutionary techniques, or even using the intelligence of the program to improve upon the intelligence process.

The most important design decision for the program is the representation method of the model. Figure 2 shows that the model can be represented as an n by m matrix where n is the number of states in the model and m is the number of input types. Each element on the matrix consists of two pieces of information, a symbol to output and the next current state. Thus, if the model is currently at state S and input symbol type B occurs, then row S and column B of the matrix defines the output response and the next current state.

At least five other pieces of information must also be maintained for the model, namely the first state, second state, the current state, value and the number of states currently in the model. The current state is used to derive the model's next response. The first state, second state and value are used by the evolution process when the model must be driven by historical data, as when determining which of the offspring during mutation is the best. The number of states currently in the machine is also used by the mutation or evolution routines, as will be seen later.

Table 1 details a workable representation of a two symbol pattern recognition model for a small computer. Each element in the n by m matrix consists of two bytes. One byte for each possible transition. For both bytes, the high order bit determines the output symbol. For example, high order bit on could mean "next number is prime," or "an earthquake will occur next," or "the opponent will next say true." High order bit off would of course mean the opposite. The other seven bits in the byte contain the relative number of the next current state, state numbers ranging from 0 to 127. Decoding the model to make a prediction is

Byte	Use
0,1	State 1
2,3	State 2
(n−1) *2, (n−1) *2+1	State n

Bit	Use
0	Output response type: on is response type 1 off is response type 0
1−7	New current state.

Table 1: The representation that is used in the computer model for the 2 symbol pattern recognition. Table 1a is the memory allocation for the matrix. Each state has two bytes with which to work. Each byte is broken into two sections as shown in table 1b. Bit 1 is used to signify the output response type, either response 0 or 1. The remaining seven bits are used to indicate the new current state.

easily done on most microprocessors and a typical sequence of instructions would be:

1. Load current state number into accumulator.
2. Shift left one place (equivalent to multiplying by 2).
3. Add input symbol type (a 0 or a 1).
4. The result of step 3 is the relative number from the beginning of the model of the byte representing the transition to be taken. Load this relative byte into the accumulator.

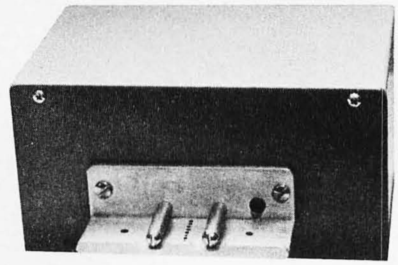
5a. If high order bit is on, then the response (prediction) is type 1.

5b. If high order bit is off, then the response is type 0.

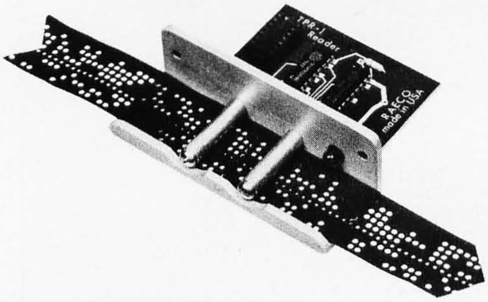
6. "AND" accumulator with hexadecimal 7F; the result is the new current state.

Part 1 of this article has described the general overall workings of the simulated evolution technique. Using the knowledge thus far presented, the reader is prepared to write his/her own artificial intelligence program for predicting earthquakes, primes or even people. Next month this discussion will continue with a description of the implementation of a predictive game with which the readers can experiment.■

What's New?



A New Low for Paper Tape Reader Prices



Personal use review by
Daniel Fylstra
Hamilton Hall C-23
Harvard Business School
Boston MA 02163

Now you can add a paper tape reader to your personal computer for only $32.50. The new RAECO TPR-1 is a "pull through" type optical paper tape reader using normal ambient light and no moving parts. A machined tape guide makes it relatively easy to pull tape smoothly through the unit. Although the reader works best with opaque (black) paper tape, it can be used with a little care to read the buff-colored tape designed for mechanical readers. The unit is easily interfaced to any processor through an 8 bit parallel IO port, with a READY (or $\overline{\text{READY}}$) signal to indicate the presence of valid data. Since the unit has three-state output buffers controlled by a separate $\overline{\text{ENABLE}}$ line, it theoretically could be connected directly to a data bus.

The TPR-1 was below my price threshold for a paper tape capability, so I ordered one for use with my KIM-1. RAECO gave me unusually prompt service, less than one week, during the busy Christmas season. Instructions supplied with the TPR-1 were clear and complete, and in less than two hours I had connected the unit to my system. After carefully positioning the desk lamp I was using as a light source, I subjected the TPR-1 to the paper tape reader "acid test," loading Tom Pittman's Tiny BASIC, a fairly lengthy program, from a buff-colored, fan-folded paper tape. Sure enough, I was able to load the program with only one retry after a checksum error.

RAECO has done an excellent job of bringing a paper tape reader capability to the personal computing user at a new low price. The RAECO TPR-1, assembled and tested, $32.50 (optional case, as pictured, $5 extra), from RAECO, Box 14, Readville MA 02137.■



RCA Has Just Published a New Product Guide to Its CMOS-B Series of Integrated Circuits

A 24 page product guide, COS-278F, *COS/MOS-B Series, The new standard in CMOS digital logic*, is now available from RCA Solid State Division. The booklet describes RCA's B series of COS/MOS high-voltage integrated circuits. All B series devices are tested to voltages as high as 22 V and have an absolute maximum DC supply voltage rating of 20 V. The comparable rating for the traditional A series is 15 V.

The product guide contains a description of the features and benefits of the B series, function classification chart, function selection chart, standardized static electrical characteristics, and function diagrams for 120 B series types, including buffered and unbuffered versions. Special information on seven standard timing circuits is also included. Copies of the COS-278F may be obtained by writing to RCA Solid State Division, POB 3200, Somerville NJ 08876.■

Zilog's Z-80A at 4 MHz

At its new manufacturing facility in Cupertino CA, Zilog announced the manufacture of the Z-80A microprocessor on February 24 1977. The Z-80A is a faster microprocessor than the Z-80 and is pin and software compatible with the Z-80. The Z-80A has a standard clock rate of 4 MHz which is made possible by a new technology developed by Zilog. Single quantity prices quoted at the press conference were $65 for the ceramic package, and $59 for the plastic package. The Z-80A will become Zilog's standard microprocessor. However, the present Z-80 at a clock rate of 2.5 MHz will be continued at a reduced price.

According to Zilog, the Z-80A is considered to be the fastest standard microprocessor. Its instruction cycle is 1 μs, and its throughput is 60% above the throughput of the Z-80. It became available in production quantities in late March, 1977. For more information write to Zilog, 10460 Bubb Rd, Cupertino CA 95014, (408) 446-4666.■

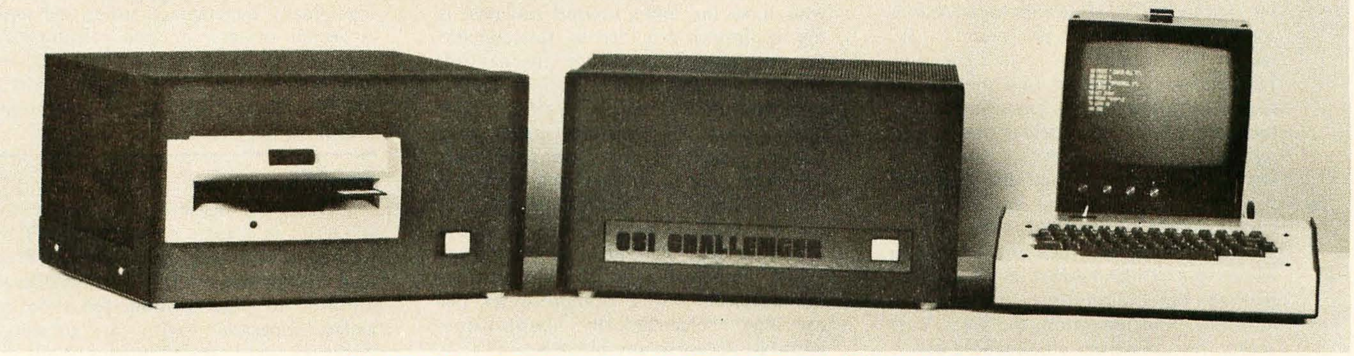
Want Megabyte Megalomania?



Here is an *inexpensive* way to achieve a 32 megabyte mass storage system for use in filing *lots* of data. The product is the Model 5200 Carousel tape cartridge system by National Computer Systems, 4401 W 76th St, Minneapolis MN 55435. What does inexpensive mean? In this context, it means that $3215 (in quantities of 50) purchases you a single Carousel engineered for use with minicomputers or microcomputers, using a built-in dual microprocessor formatter which provides interfacing disciplines, handles housekeeping functions for the storage device, and reduces the user computer's driving software requirements to command information and data transfer.

This machine is designed for unattended operation. All the mechanical operations of indexing the cartridge positions, loading, processing data and unloading are done automatically. This means that data can be copied, files sorted, information updated and reports

INTRODUCING THE COMPLETE CHALLENGER SYSTEM.



OSI's DREAM MACHINE
is a totally integrated computer system.

Imagine a system complete with terminal, CPU, memory, floppy disk, software, and all the little necessities to make it work together immediately. Now imagine this complete system available not only fully assembled, but priced much lower than anyone else's kit. What you are dreaming of is OSI's "new" Challenger System!

In the configuration shown above, the Challenger includes everything an end user needs for a complete small computer system. All you add is 110 VAC power and a desk to put it on.

This fully-assembled system includes:

HARDWARE:

OSI Challenger 65 with 16K RAM, serial interface, system monitor PROM, and floppy disk bootstrap PROM.

OSI Challenger single drive floppy disk formatted for 250K bytes storage per diskette surface.

Stand-alone terminal and Sanyo monitor for 16 lines of 64 characters at 2400 baud (other terminal options are available).

And all interconnecting cables!

SOFTWARE:

2 diskettes containing over 100,000 bytes of software including OSI's powerful Disk Operating System with variable length sectors. **6502 DISK BASED RESIDENT ASSEMBLER/ EDITOR!** A totally interactive Assembler/Editor which assembles up to 600 lines a minute and is completely compatible with MOS Technology's Cross Assembler format. This program also contains a powerful disk-based line editor with commands for general text editing. **OSI'S EXTENDED MONITOR:** A powerful machine language debugging and utilities package including a Disassembler which is format compatible with the Assembler! **OSI 6502 8K BASIC FOR DISK BY MICROSOFT:** This powerful BASIC has all the features of Altair® 8K BASIC for the 8080 plus higher speed and disk storage. And it comes complete with a BASIC program library.

DOCUMENTATION AND SUPPORT:

We include over 600 pages of hardware, software, programming, and operation manuals. The Challenger is based on the well-proven OSI 400 system. The over 2,000 OSI 400s and Challengers now in use assure continuing hardware and software support for this system for years to come!

EXPANDABILITY:

The Challenger System can now be expanded to 192K of RAM and 16K of I/O and ROM. There are over 13 accessory boards including A/D, D/A, parallel and serial I/O, cassette interfaces, a dual drive floppy, a video graphics display, several RAM and PROM boards, and multiple-processor configurations.

APPLICATIONS:

The Challenger system is complete, fully assembled and configured so that the Disk Operating System can be booted in immediately on system power-up. Even a relatively inexperienced operator can have a complex BASIC program on-line just seconds after the system is turned on. The ease of use, high reliability, and large library of standard BASIC applications programs make the OSI Challenger System the first practical and affordable small computer system for small business, educational institutions, labs, and the personal computerist.

PRICES:

Challenger System, complete as stated above with terminal and monitor

$2599.00

As above without terminal. Specify RS-232 or 20ma loop and baud rate

$2099.00

IMPORTANT NOTE:

One of the most important features of the Challenger System is that it is not really "new". OSI has been delivering the basic circuitry of the Challenger since November 1975 and the floppy disk since June 1976. The only thing new is the total integration of the components as a complete, simple to use, fully-assembled, small computer system.

For more free information and the address of the OSI Computer Dealer or representative in your area, write to: OSI; Dept. S; Hiram, Ohio 44234 or enclose $1.00 for the full OSI catalog which contains kits from $134 and fully assembled computers from $439.

OSI

Ohio Scientific Instruments

11679 Hayden Street, Dept. S, Hiram, Ohio 44234

spooled without operator intervention. Put this device together with an LSI-11, DEC's timesharing operating systems, multiple terminal ports, and a school or business will have a very inexpensive data processing system with lots of usable mass storage. Usable in this case means 8000 bytes per second data transfer rate to the 3M style cartridges, search speed of 120 inches per second (3.1 meters per second) and rewind speed of 180 inches per second (4.6 meters per second). This translates (assuming a 300 foot (91 meter) 3M cartridge tape capacity) into a maximum access time of 20 seconds for any given cartridge, assuming that you are at the last block on that cartridge and you have to rewind to get to the first block. Indexing from cartridge to cartridge takes 4.25 seconds, so making the assumption that we can index in either direction, the worst case access time to a given block would be:

Rewind from last block, present cartridge	20 seconds (180 ips, 300 feet)
Step eight times 4.25 x 8 =	34 seconds
Search forward to last block, new cartridge	30 seconds (120 ips, 300 feet)
	84 seconds

Does This Have Any Relevance to the Individual?

Yes and no. Yes, in the fact that for some individuals, the price is well within budget. We all know of people who have full fledged minicomputer systems as their personal computers. One example is a data processing vice president of a large East Coast bank who has a PDP-11/40 in his home with hard surface disks and the whole works. The single quantity price of this Carousel drive will most likely fit within the "conceivable" personal computing budget of the high level management person such as this fellow (or owner of a small business with potential commercial applications).

But more important, to those of us who have far more limited budgets, is the inspiration provided by this concept of the multiple cartridge mechanism. Why not do the same thing with Philips cassettes for less money, perhaps with fewer units of the media? Or even consider a simpler mechanism with fewer cartridges used by the personal computer experimenter on a lower budget? Just the very concept of a 3M drive with the 1/4 inch tape cartridges is enough to excite the personal user, since it provides the potential for 2 million bytes per cartridge, electronically controllable with fast access to any block.■

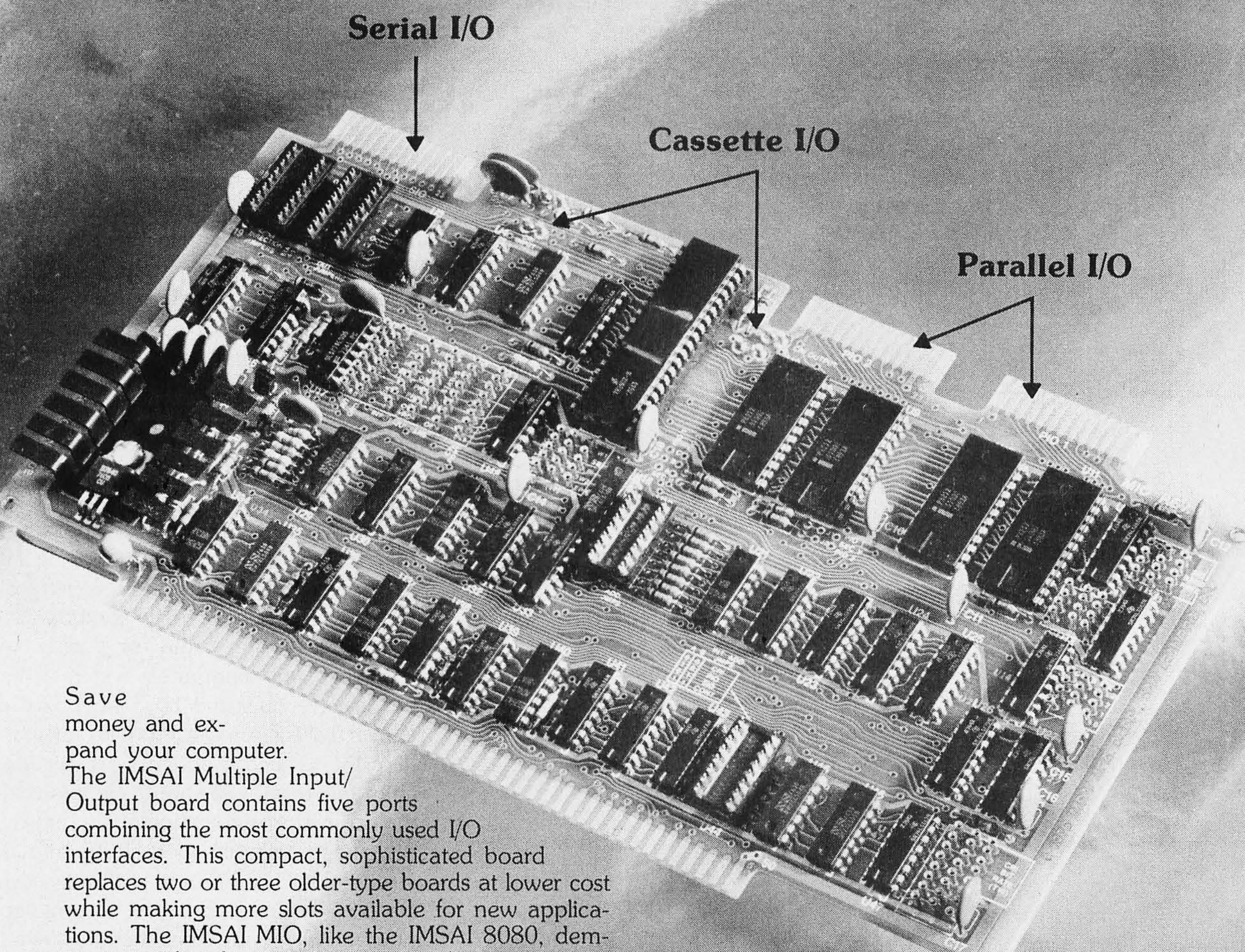
Expansion of Capacity

Latest word from Texas Instruments in Dallas is announcement of an increase in the programmable memory and data capacity of the SR-60 desk top calculator. This machine is now available with up to 5760 steps of programming with 430 data registers — almost enough capacity to wonder whether or not it is really a calculator as opposed to a full fledged computer. For all that capacity in the new form, you'll pay a bit however: $3,315 is the total price for an SR-60 equipped with that many program steps and data registers. Texas Instruments calculator products are available from many department stores and similar retail outlets, or you can forward inquiries to Texas Instruments Inc, Inquiry Answering Service, POB 5012, MS 308, Dallas TX 75222, attention: SR-60 Memory Addition.■

TARGET

Altair Program Library entry #5-3-764 is a game called TARGET by George W Rompot, which uses the Processor Technology VDM-1 video display monitor to play a simulated shooting gallery operation with a "gunship," "missile" and "target" moving about the screen. The program is written in 8080 machine code. The target moves at 5 lines per second up and down the right edge of the machine, and the missile moves across the center of the screen at 120 character positions per second; the missile is fired in response to a keyboard input of any character. The skill required is represented by the need to "lead" the target. The MITS game comes in the form of 9 pages of documentation including 5 pages of commented program listing in octal for this 197 byte program and a 13 byte screen initialization routine.■

IMSAI Presents:
The 3 for 1 Slot Saver
A Masterpiece in I/O Engineering.



Save money and expand your computer. The IMSAI Multiple Input/Output board contains five ports combining the most commonly used I/O interfaces. This compact, sophisticated board replaces two or three older-type boards at lower cost while making more slots available for new applications. The IMSAI MIO, like the IMSAI 8080, demonstrates that fine engineering can be aesthetically beautiful as well as solidly practical.

USEFUL AND EASY TO USE.

The IMSAI MIO is inexpensive and easy to install. It's the most advanced I/O board available, adding quality, convenience and value to your computer. With the IMSAI MIO you can control a keyboard, printer, 2 tape cassettes and a teletype/CRT simultaneously.

Consider these important "useability" features:

1. Easy to assemble components — all board identities are silk-screened.
2. Safe soldering — the entire board is solder masked for quick, fast assembly.
3. Fast, flexible configuration — options are selected with solderless wire strapping.
4. Easy testing and tuning — pre-recorded test routines on tape cassette.
5. Best of all — everything is explained in comprehensive, understandable documentation.

TOTAL FLEXIBILITY.

A control port, serial channel, 2 parallel I/O ports and a two-tape cassette interface provide all the versatility you need in an I/O board. The IMSAI MIO's flexible addressing and control features assure compatibility with non-IMSAI 8080 software. Tape cassette applications are enhanced through inclusion of the Byte/Lancaster and Tarbell recording modes.

ORDER NOW—SHIPPING NOW

The IMSAI MIO is available for immediate delivery. $195 in kit; $350 assembled. Write or call for more information. Specifications and feature product bulletin available on request. Send $1 for complete catalog of IMSAI products.

Prices: USA Domestic. Subject to change without notice.

IMSAI

IMSAI Manufacturing Corporation
14860 Wicks Blvd.
San Leandro, CA 94577
(415) 483-2093
TWX 910 366-7287

Come Upstairs

Steve Ciarcia
POB 582
Glastonbury CT 06033

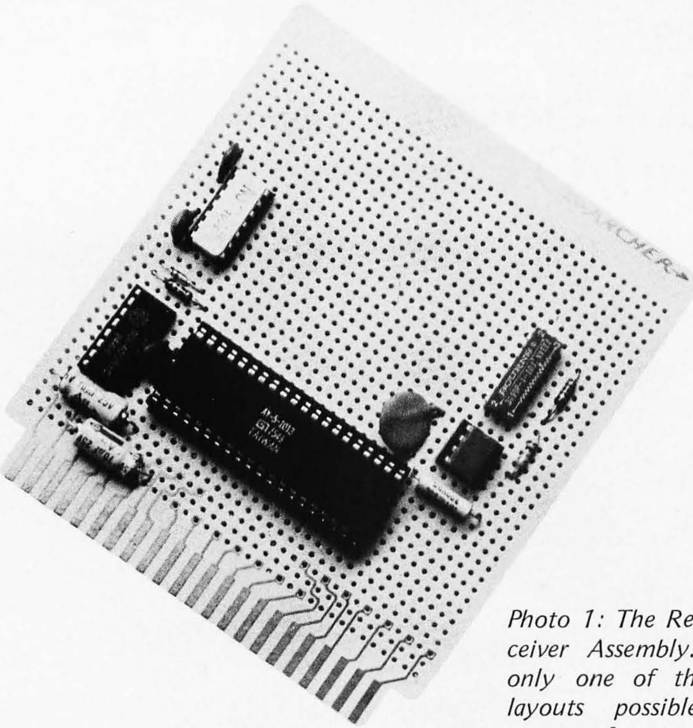


Photo 1: The Remote Receiver Assembly. This is only one of the circuit layouts possible. Placement of parts is not crucial. The receiver is built so that it can be plugged directly into the microprocessor's main board.

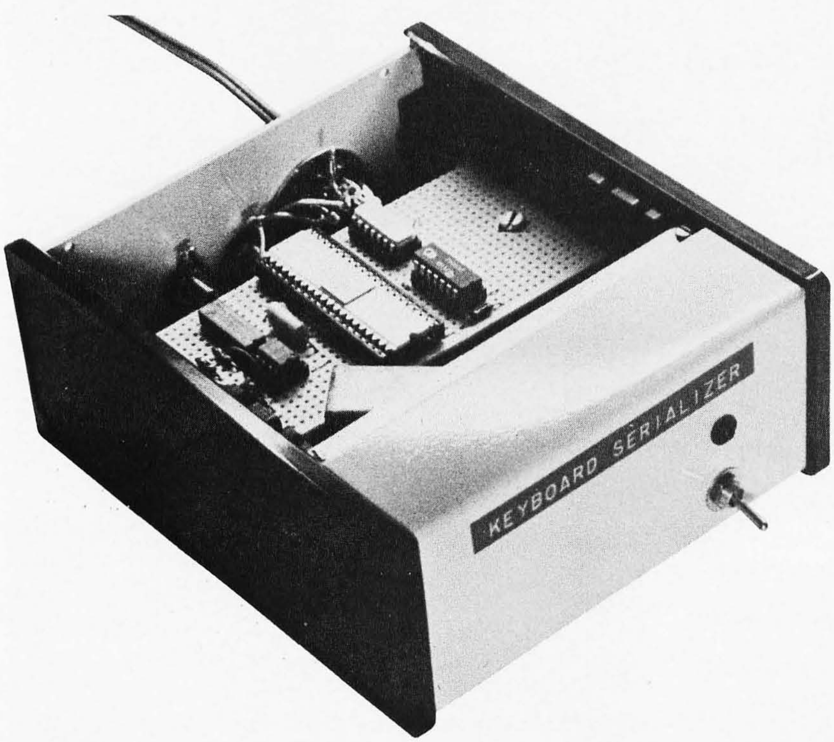


Photo 2: The Remote Transmitter Assembly. The transmitter also needs a power supply, contained within this box, for both itself and the keyboard. With a stylish cabinet such as this, you can be proud to display your newest addition to your friends.

For those of you who remember about Walt and Ralph in "Having a Private Affair With Your Computer" *[April 1977 issue of BYTE]*, I suppose I can say that with friends like that I am not in any great need of enemies. What may have appeared as a losing situation on my part did have some beneficial side effects – namely, the remote terminal which I installed in my den to accommodate the comings and goings of people using my system.

This remote terminal is nothing more than a serial data link incorporating a universal asynchronous receiver-transmitter (UART) attached to a keyboard and a coaxial extension cable for a direct plug-in monitor. Any computer system which can be directly operated by a keyboard can be converted for remote input exactly as I will describe. The components are readily available and total cost should be less than $30.

The major application for a remote interface is to be able to describe and demonstrate your computer without having to clean up your shop before letting people in or worrying about what they'll touch. Often, when I have large parties, I will put a Kingdom game on the computer and let people play to their hearts' content in the den. Obviously, trying to do this in the basement where the computer is located would require constant attendance. All it would take is one drunk to lean on the processor card, or say, "What does this do?", as he flips the memory clear switch. I will have to admit though that what had originally started as a purely defensively initiated design has become an enjoyable addition to my system. If just writing software, I'll do it upstairs now, with the family, rather than appearing to isolate myself in the cellar every night, getting moldy, as my wife says.

A simplified block diagram of this remote terminal link is illustrated in figure 1. For the keyboard inputs to the computer both transmitter and receiver interfaces utilize an interesting device called a universal asyn-

and Be Respectable

chronous receiver-transmitter. The internal structure of this device consists of a separate parallel to serial transmitter and serial to parallel receiver joined by common programming pins. What this means, in fact, is that the two sections can be used independently provided they adhere to the same bit format which is selectable through the wiring of the device.

The transmission from the den to the basement is done asynchronously and in one direction only. As far as the computer is concerned, this input device appears as any other parallel input such as a keyboard. The other part of the system is a monitor to display computer generated video. The monitor in the den uses direct video signals and is attached to the video display electronics through a 200 foot coaxial cable, and thus does not require the computer to respond via the serial line.

Actual data transmission to the computer follows an asynchronous serial format illustrated in figure 2. When no data is being transmitted, the data line is sitting at a mark or 1 level waiting for a key pressed strobe from the keyboard. A key pressed strobe is a 1 to 5 ms positive pulse indicating that a keyboard key has been pressed and that an ASCII code of that key is available for transmission. This key pressed strobe, which is attached to the data strobe of the interface, causes the ASCII data to be loaded into a parallel storage buffer, and starts the transmission cycle. The serial output will then make a transition from a logical 1 to 0. This start bit transition indicates the beginning of a serially transmitted word. Following the start bit, up to eight bits of data follow, each data bit taking 16 clock periods. At the conclusion of the data bits, parity and stop bits are generated by the interface to signify the end of transmission.

If another key is pressed, the process will repeat itself. On the receiving end, the receiver section of the UART is continuously monitoring the serial input lines for the start bit. Upon its occurrence the

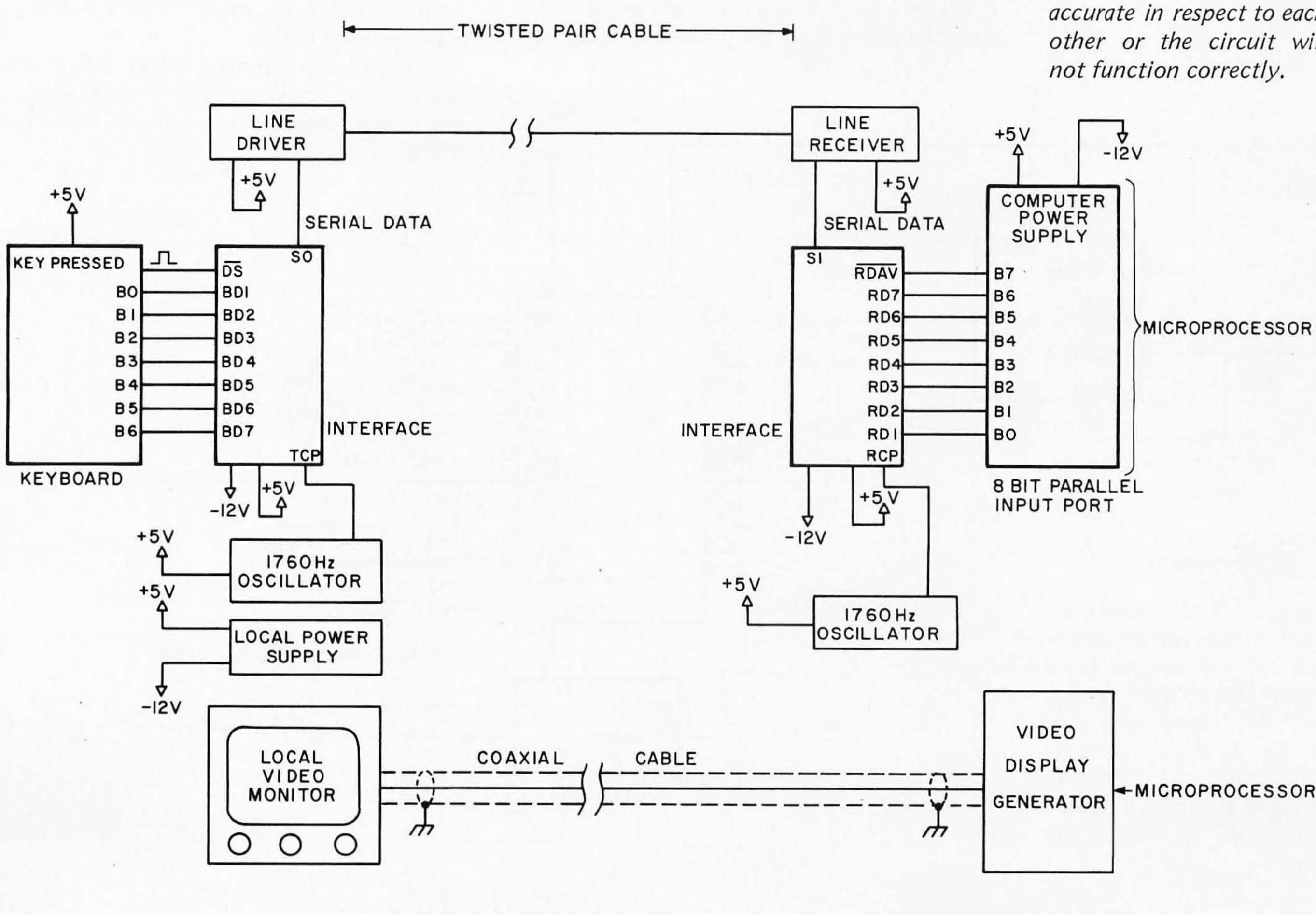


Figure 1: A simplified block diagram of the remote transmitter and receiver circuit. With a 1760 Hz oscillator driving the circuit, a data transmission rate of 110 bps will be attained. Although the exact frequency is not important, the two oscillators must be accurate in respect to each other or the circuit will not function correctly.

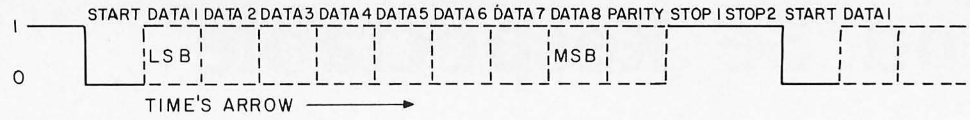


Figure 2: Diagram illustrating how asynchronous data is transmitted. A start bit is defined as a transition from logical 1 to 0. The eight data bits are then transmitted with the least significant bit being transmitted first and the most significant bit being transmitted last. A parity bit is then sent followed by two stop bits. With the AY-5-1013, the parity bit can be omitted completely and there is a choice of one or two stop bits following the transmitted data.

eight bits of data are slipped into a register and the parity checked. At the completion of the serial entry, an output signifying data available is set which can be used as an input strobe to the computer. The interface will not process additional serial inputs unless the data available flag is acknowledged and the data available reset line is strobed. Actual transmission can include or exclude parity, have one or two stop bits, and data can be in 5 to 8 bit words. These options are pin selectable.

Table 1 is a pin function description for the AY-5-1013.

There are obviously many more functional inputs and outputs than are necessary for this limited application. We will purposely select only those which are absolutely necessary to minimize hardware and software logic. The keyboard has a 7 bit ASCII output code so a 7 bit word length is chosen by a binary 10 code on pins 37 and 38 respectively. Two stop bits provide for a more reliable transmission by allowing more time between transmitted words. Two stop bits are selected by tying pin 36 to +5 V. Parity is nice, but unless you are going to do something about it, it's a waste of time. At 110 bps, failures will be rare and in many hours of operation I have not yet received a bad character transmission. Parity is eliminated by tying pin 35 to +5 V.

Both transmitter and receiver interfaces utilize an NE555 oscillator to provide the clock frequency (see figures 3 and 4 for schematics). For a 110 bps serial rate, this frequency is set for 1760 Hz and must be maintained within 1%. For this reason only mylar or polycarbonate capacitors should be used. This transmission rate can be much higher, possibly reaching a rate of 20 or 30 thousand bits per second. But we must be realistic: 110 bps is 10 characters a second, which is beyond the typing capabilities of most sane individuals. Building a 9600 bit per second data link to a keyboard is absurd. The lower the data rate, the fewer problems you will encounter in the long run.

Construction of the receiver and transmitter boards is straightforward and wiring is not critical. The unit in the den requires a +5 V and -12 V power supply and an appropriate case. Most new keyboards require +5 V for operation and the interface power supply should be made large enough to accommodate both.

All my keyboards have cables with 11 pin "octal type" sockets *[electron tube sockets and plugs similar to those of the large*

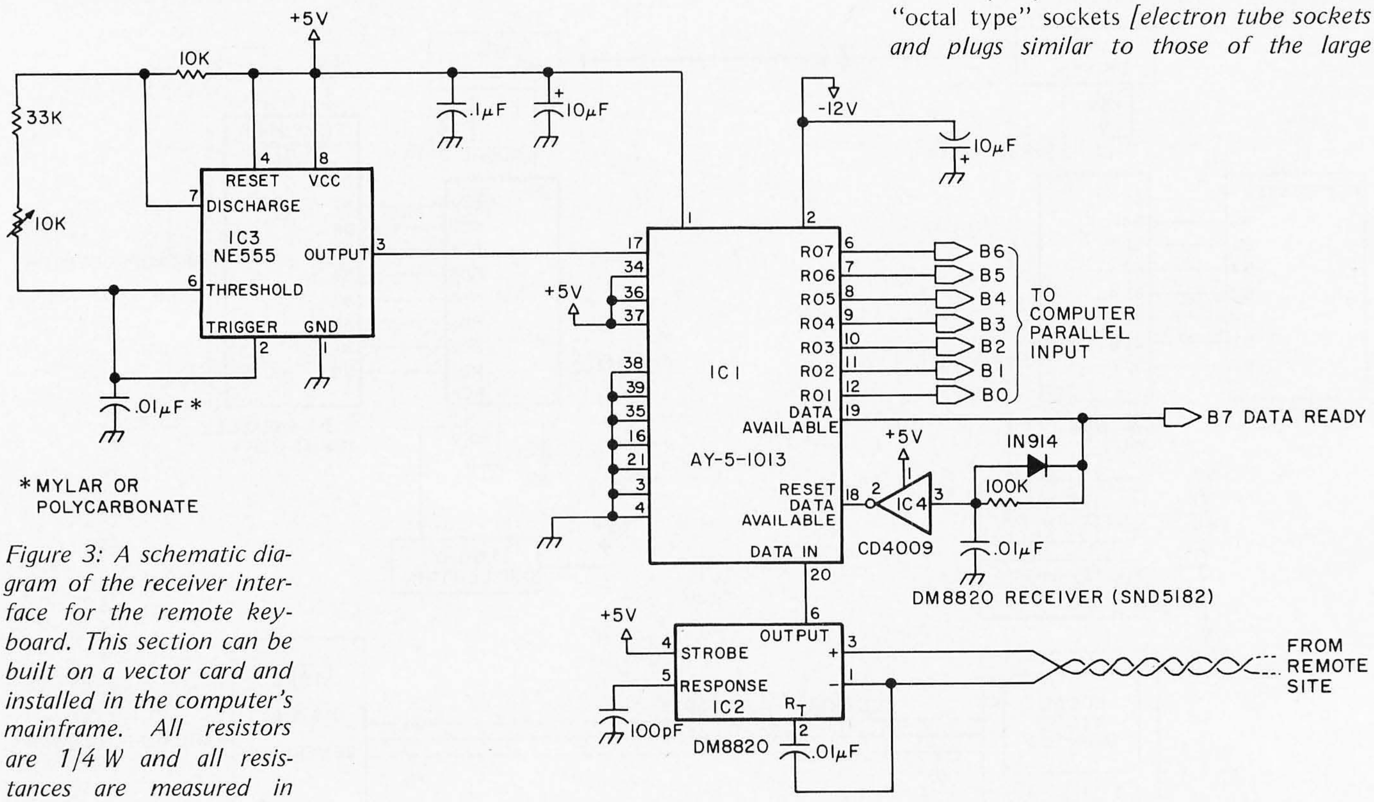


Figure 3: A schematic diagram of the receiver interface for the remote keyboard. This section can be built on a vector card and installed in the computer's mainframe. All resistors are 1/4 W and all resistances are measured in ohms.

"octal" tubes], so I naturally used the same receptacle on this transmitter box. The cable and connector can be any convenient type, and the keyboard will probably pull less than 200 mA. The only other connection is the twisted pair wire to the receiver board and this should be some easy disconnect like an RCA jack or two banana jacks, as I used.

The receiver construction is equally simplistic and its proximity to the computer eliminates the necessity for building additional power supplies. When a character is

Pin Number	Name	Symbol	Function
1	VCC power supply	VCC	+5 V power supply.
2	VGG power supply	VGG	−12 V power supply.
3	VDD power supply	VDD	ground.
4	$\overline{\text{Received Data Enable}}$	$\overline{\text{RDE}}$	logic 0 places the received data onto the output lines.
5–12	Received Data Bits	RD8-RD1	eight data output lines. Received characters are right justified; the least significant bit always appears on RD1. These lines have three state drivers enabled by a low level on the $\overline{\text{RDE}}$ line.
13	Parity Error	PE	goes to logic 1 if received character parity does not agree with selected parity. Three state output enabled by $\overline{\text{SWE}}$.
14	Framing Error	FE	goes to logic 1 if the received character has no valid stop bit. Three state output enabled by $\overline{\text{SWE}}$.
15	Over Run	OR	goes to logic 1 if previously received character was not read before present character is transferred to receiver holding register. Three state output enabled by $\overline{\text{SWE}}$.
16	$\overline{\text{Status Word Enable}}$	$\overline{\text{SWE}}$	logic 0 on this line places status word bits PE,FE,OR,DAV,TBMT onto the output lines. Three state outputs.
17	Receiver Clock	RCP	contains clock whose frequency is 16 times the desired receiver data rate.
18	$\overline{\text{Reset Data Available}}$	$\overline{\text{RDAV}}$	logic 0 will reset the data available line.
19	Data Available	DAV	goes to logic 1 when an entire character has been received and transferred to the receiver holding register. Three state output.
20	Serial Input	SI	accepts the serial bit input stream. A logic 1 to logic 0 transition is required for initiation of data reception.
21	External Reset	XR	resets shift registers. Sets SO,EOC,TBMT to logic 1. Resets DAV and error flags to logic 0. Clears input data buffer. Tied to logic 0 when not in use.
22	Transmitter Buffer Empty	TBMT	transmitter buffer empty flag goes to logic 1 when data bits holding register may be loaded with another character. Three state output enabled by $\overline{\text{SWE}}$.
23	$\overline{\text{Data Strobe}}$	$\overline{\text{DS}}$	strobe on this line will enter data bits into the data bits holding register. Initial data transmission is initiated by the rising edge of $\overline{\text{DS}}$. Data must be stable during entire strobe.
24	End of Character	EOC	goes to logic 1 each time a full character is transmitted. It remains at this level until the start of transmission of the next character.
25	Serial Output	SO	serially, by bit, provides the entire transmitted character. It will remain at logic 1 when no data is being transmitted.
26–33	Data Bit Inputs	BD1-BD8	eight data bit input lines.
34	Control Strobe	CS	logic 1 enters the control bits EPS, NB1, NB2, TSB, NP into control bits holding register. Line can be strobed or hard wired to logic 1 level.
35	No parity	NP	logic 1 will eliminate the parity bit from the transmitted and received character. Stop bits immediately follow the last data bit. If not used, must be tied to logic 0.
36	Number of Stop Bits	TSB	selects the number of stop bits, 1 or 2, to be appended immediately after the parity bit. A logic 0 will insert one stop bit and a logic 1 will insert two stop bits.
37–38	Number of bits per character	NB2,NB1	two leads internally decoded to select five, six, seven, eight data bits per character. NB2 / NB1 / Bits per character 0 / 0 / 5 0 / 1 / 6 1 / 0 / 7 1 / 1 / 8
39	Odd or even parity select	EPS	determines type of parity appended immediately after data bits. Logic 0 inserts odd parity, logic 1 inserts even parity.
40	Transmitter Clock	TCP	contains a clock whose frequency is 16 times the desired transmitter data rate.

Table 1: A summary of the pins and function description of the AY-5-1013 universal asynchronous receiver and transmitter.

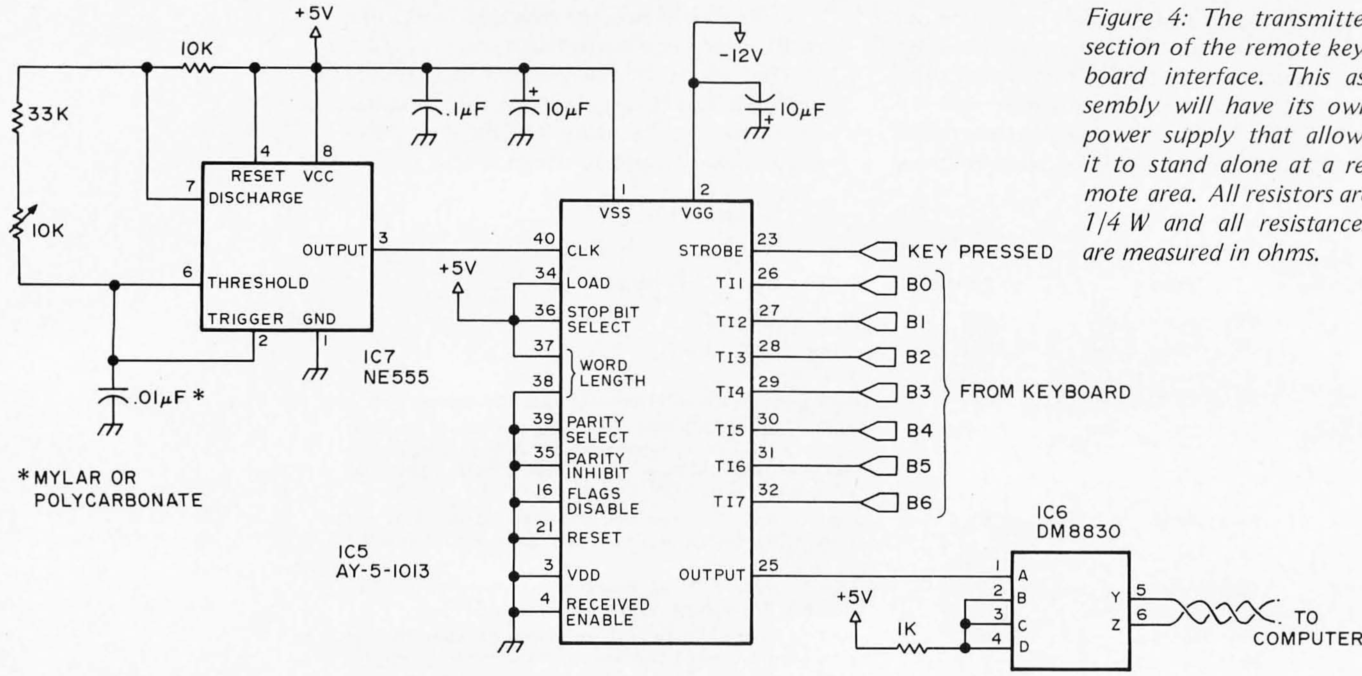


Figure 4: The transmitter section of the remote keyboard interface. This assembly will have its own power supply that allows it to stand alone at a remote area. All resistors are 1/4 W and all resistances are measured in ohms.

received, the data available line goes high. This signal is delayed and inverted and used to reset the data available line. The result of this circuitry is that the data available line will remain high only for the duration of the time delay set by the resistor and capacitor at the input of the CMOS inverter. If your system will not accept a 10 ms data ready strobe, then dispose of the CMOS gate and reset pin 18 via a computer output strobe or some other synchronized means.

The last point of discussion is the line driver and receiver. RS232C seems to be the standard, but this would require a plus and minus 12 to 15 V power supply for the drivers and receivers. The limits of most microprocessor power supplies are usually within +12 V and -12 V. Let's be practical for a moment. We aren't stretching a 1 mile cable along an arc welding line at an automobile plant (as occasionally happens when working with industrial automation as I do for a living). What we have is 100 to 200 feet of twisted pair in a relatively low noise environment. RS232C is for industrial environments, and is not necessary for this application. 5 V line drivers are quite acceptable in this application, and a 7440 NAND gate could be used as an appropriate driver. A somewhat preferred approach is to use 5 V differential drivers and receivers such as the DM8820 and DM8830. These National Semiconductor (or their Texas Instruments equivalents SW75182 and SW75183) devices may not be available everywhere. A Texas Instruments 75107 and 75110 combination with the appropriate pin numbers changed can be used instead.

Construction is straightforward, and checkout is a simple go or no go test. Make sure that all voltages are correct and that the keyboard inputs to the interface operate correctly. It is a good idea to use a frequency counter to set the 1760 Hz on both the transmitter and receiver cards. If the frequencies are not within a few percent of each other, it will not work.

Attach a 1 or 2 foot twisted pair lead between the two units, plug in the keyboard, and remove the CMOS inverter from the receiver card. To display the received data word, either connect the receiver output to a computer parallel input port and write a program to scan and display it, or attach some sort of LED buffer and driver kluge to display the output.

When a key is pressed on the keyboard, the ASCII code for that key should appear at the receiver output. Momentarily grounding the data available reset line will allow another character to be received. When this phase is accomplished, insert the 4009, string your 200 feet of twisted pair cable, hook your monitor up with 200 feet of coax, and start plinking away.■

IC Number	Type	+5 V	GND	−12 V
1	AY-5-1013	1	3	2
2	DM8820	14	7	
3	NE555	8	1	
4	4009	16	8	
5	AY-5-1013	1	3	2
6	DM8830	14	7	
7	NE555	8	1	

Table 2: Power pin assignments for the receiver and transmitter of figures 3 and 4.

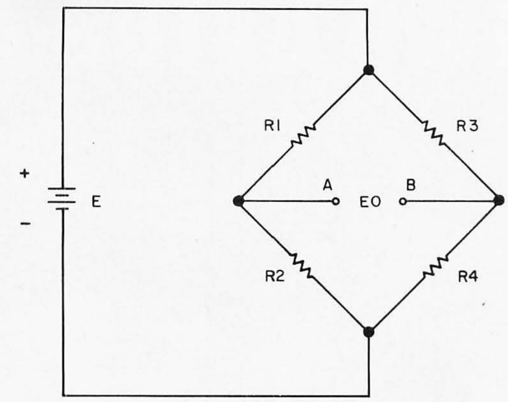


FREE YOURSELF FROM THE ONE BYTE WORLD. MOVE UP TO THE TWO BYTE **TEXAS INSTRUMENTS TMS-9900 16-BIT MICROPROCESSOR** — WITH OUR — **"SUPER STARTER SYSTEM"** — TEC-9900-SS. SHOWN ABOVE, FEATURES HARDWARE MULTIPLY AND DIVIDE, 69 MINI-COMPUTER INSTRUCTIONS, 7 ADDRESSING MODES, EXPANDABLE TO A FULL 65K BYTES; MONITOR, TMS 9900 CPU, RAM, P-ROM, E-PROM, PROGRAMMER ALL ON ONE P-C BOARD BASIC OPERATING SYSTEM AS LOW AS **$299 UNASSEMBLED $399 ASSEMBLED AND TESTED** EXPLICIT **MANUAL** INCLUDED OR AVAILABLE SEPARATELY AT **$35,** TO LEARN MORE . . .JUST TEAR OFF A PIECE OF THIS AD, PIN TO YOUR LETTERHEAD & RETURN TO TECHNICO OR **CALL OUR HOTLINE 1-800/638-2893** TO RECEIVE FREE INFO-PACKAGE. --- DESIGN & TECH SUPPORT BY ROSSE CORP.

TECHNICO INCORPORATED

9130 Red Branch Rd.
Columbia, Md. 21045
301-596-4100

THE TECHNOLOGY LEADER IN ELECTRONICS DISTRIBUTION



Joseph J Carr
Bioelectronics Laboratory
The George Washington University Medical Center
901 23rd St NW
Washington DC 20037

Interfacing With an

The jobs which a computer can perform require some method for interfacing with the outside world. Most of us are familiar with the requirements and abilities of the many digital peripherals, but lack any practical insight into the use of analog IO signals. To many, especially those oriented toward software, doing anything in the analog world seems like an anachronistic art form, or from a technical point of view, a lot like trying to skin an amoeba. However, the analog world exists, and some means must be provided for interfacing the computer with it. Since most devices and machines in our world are analog, this will extend your computer horizons more than you may appreciate. So if you want to put your machine to the task of some really serious work, the time has come to learn about some of the requirements of analog electronics.

Several problems present themselves in this area. We must understand something of the measurement of physical and electrical parameters from the real world, amplification and signal processing, analog readout devices, and conversions from analog to digital or vice versa.

Transducers

A transducer is any device which transforms a measurement of a parameter from one form to another for purposes of measurement or control. We will modify this definition a little to also specify that, for our purposes, the parameters we measure are converted to electrical signals. We will also consider devices which are merely electrodes to pick up existing potentials, as in biomedical applications, and also those which will only produce a switch contact closure if certain conditions are met. An example of the latter is the thermostat in your home.

Transducers are available for almost any type of physical parameter or phenomena. Assorted manufacturers offer devices that will produce an output voltage that is proportional to physical parameters such as force, strain, pressure, temperature, light, position, flow or a host of other variables.

The function of a transducer leads us to understanding just what we mean by analog electronics. Most people will tell you that an analog signal is one which is constantly varying, as opposed to digital signals which may have but two states. A more reasonable definition also takes into account that the voltage or current input to the circuit is an electrical analog of some physical parameter being measured. In this case, the word analog is related to analogous.

The number of different types of transducers prevents us from being too specific in this context, so we won't even try except to point out that most transducers which you will encounter will be resistive and have a Wheatstone bridge circuit such as is shown in figure 1. The voltage, E, must be regulated in order to prevent variations in the excitation

Figure 1: A basic resistive transducer in Wheatstone bridge form. The circuit produces an output voltage that follows the relationship:

$$E_O = \frac{R2}{R1 + R2} - \frac{R4}{R3 + R4}$$

The excitation voltage E is usually between 0 V and 10 V for most common forms of transducers. It may be either AC or DC, but DC types may be preferred for simplicity.

Analog World—Part 1

potential from showing up as artifacts in the transducer output signal. I have used LM309 and 7805 three terminal integrated circuit regulators for this function with good success.

Determining the Resistors for the Right Job

In most transducers of the class which we are considering, resistors R1 through R4 will change resistance with changes in the applied parameter being measured. Although some transducers, notably temperature types, will have only one or two resistors of the Wheatstone circuit variable, most will allow all four to vary with the applied stimulus. Those resistors which do not vary are fixed, of course.

The nature of the resistors in the bridge will depend totally on the type of phenomena or parameter being used to stimulate the transducer. Thermistors, for example, are appropriate if it is temperature that is being measured. Piezo-resistive strain gauge elements, on the other hand, may be more appropriate for measurement of force, pressure and so forth. A simple linear taper potentiometer may be used to measure position. For the casual designer with but moderate requirements, the only specifications of real interest are such items as the maximum and minimum range, linearity and sensitivity.

The range is the maximum and minimum values of the applied parameter which can be accommodated. A pressure transducer, for example, might be specified over a range of −100 Torr to +400 Torr (1 Torr = 1 mm Hg). It is necessary to know just what the manufacturer of the transducer means when range specs are given. The figures quoted might be destruction points or they might be the range over which other specs, such as linearity and sensitivity, are guaranteed. In either case, though, the range limits should not be exceeded.

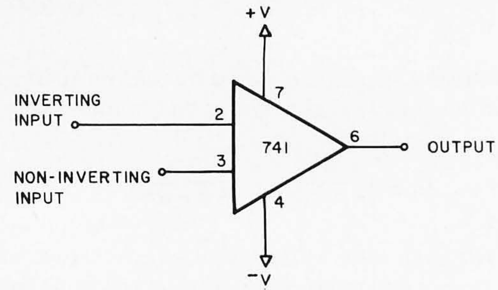
Desirable Qualities

Linearity is a measure of how well the transducer output follows variations in the applied parameter. This is usually quoted as a percentage of either full scale, or some test point, at which the device is calibrated. Linearity can be thought of as a measure of how good the transducer is – in other words, how analogous the output voltage is to the input stimulus. You will find that, in general, linearity can be expensive; the higher the cost, the better the linearity.

Sensitivity is one specification of primary interest to the instrument designer. One method for denoting this specification is in terms of transducer output voltage per unit of applied stimulus per volt of excitation potential. This might be stated as:

$$S = X\ \mu V/V/\text{unit stimulus}$$

Figure 2: The basic figure for a 741 operational amplifier with power connected. The inverting terminal (—) produces an output signal that is 180° out of phase with the input voltage. The noninverting input terminal (+) produces an output that is in phase with the input. It should be noted that the pinouts shown are for the 8 pin dual in line (DIP) and metal can packages. This particular circuit also comes in dual in line packages with two or four amplifier sections sharing common power supply pins.



A pressure transducer, for example, might be rated at a sensitivity of 100 μV/V/10 Torr. Assuming the common excitation potential of 5 V DC, this tells us to expect an output voltage of:

$$100\ \mu\text{V/V/10 Torr} \times 5\ \text{V} = \frac{50\ \mu\text{V}}{\text{Torr}}$$

A pressure of 100 Torr, then, will produce a voltage of:

$$\frac{50\ \mu\text{V}}{\text{Torr}} \times 100\ \text{Torr} = 5000\ \mu\text{V, or } 5\ \text{mV}$$

Analog Signal Processing

The processing of analog signals is one area where analog and digital buffs are frequently at dissonance with each other. To be sure, many of the functions of analog signal processing can be done equally well in either field. In some areas there may be a clear bias in favor of either analog or digital approaches.

The normal signal processing functions may be considered to be amplification, filtering, compression and so forth. Amplification seemingly becomes easy in software because it is merely necessary to multiply the input voltage, represented by a digital word, by some constant called the amplification factor. This is less easy than may appear at first blush because the input might be too low to take full advantage of the analog to digital converter's range.

Filtering is an area where software signal processing has really come into its own. In fact, it is possible to design a software filter that is a lot better than all but the most costly hardware analog filters. The problem comes in trying to implement the software filter in a limited amount of memory. On systems with a small amount of memory capacity, say less than 1 K bytes, the hardware filter might be more useful.

Another aspect of signal processing is the derivation of parameters. For simplicity of hardware design it is sometimes easier to measure one parameter than another. With this in mind, the actual electronics can often be kept simple prior to the analog to digital conversions. It is often possible to use software to do unit conversions and compute derived parameters from related, but easier to measure, parameters.

Amplifiers

A computer will want to see an input in the form of a digital signal flag or an n bit word. Since the analog instrument may only produce a voltage or current level this presents a problem. In that event a suitable converter is required. It must be noted, though, that many modern digital instruments used in science and engineering, especially those with digital readout, will provide rear panel digital output, usually in BCD (binary coded decimal) form.

Most common analog to digital converters accept full scale input signals over a limited range, for example, between ± 5 V, 0 to +10 V, or 0 to +5 V. The typical transducer, on the other hand, may want to output only a few millivolts or less, a level that may well be in the noise as far as the analog to digital converter is concerned. In such cases, it would be necessary to provide intermediate amplification.

Although numerous approaches to amplification exist, the integrated circuit operational amplifier is probably most applicable by the largest number. In fact, many small computer systems already have the ± 12 V power supplies usable for operational amplifier chips. In most of the applications which you will find, the familiar 741 type operational amplifier is probably the best bet. It comes in several package styles and has two endearing qualities: It is forgiving of novice designers and it is quite inexpensive, typically less than $1.

It is obvious in the limited context of this article that I cannot fully describe either the operational amplifier in its glory, or even the full spectrum of its uses. For those interested let me refer you to my book *Op-Amp Circuit Design and Applications* (Tab Books #787). For this present discussion we shall limit ourselves to the basic rules governing operational amplifier circuits of use to you.

The symbol for the basic operational amplifier can be seen in figure 2. Note that there are two different power terminals and that neither is normally grounded. Although you will see some circuits in which one

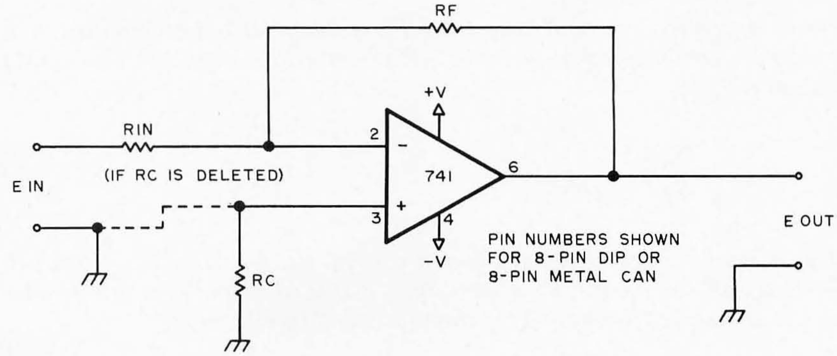


Figure 3: This circuit is a simple inverting amplifier. It will produce an output voltage according to the relationship:

$$E_{out} = -A \times E_{in}$$

The voltage gain factor A is equal to the ratio R_f/R_{in}. There are restrictions on the value of this ratio which must be met in order for the operational amplifier to function properly (see text).

power terminal is grounded, in normal operation a symmetric dual power supply is used. The +V terminal is always connected to a supply that is more positive than the −V terminal however the circuit is configured. Symmetric potentials appropriate for most operational amplifier integrated circuits now available range between ± 4 V and ± 15 V. Certain special or premium devices extend the limits on each end by a couple of volts.

Simple Inverting Gain Follower

An ideal operational amplifier would allow universal application with any gain factor (see figure 3), but real operational amplifiers impose certain constraints on the designer. For example, it is not wise to allow the value of R_{in} to drop below 5 to 10 times the impedance of the transducer or other driving source. A transducer might, for example, have an effective impedance (resistance) of 100 ohms. In this case, the minimum acceptable value for R_{in} would be 1000 ohms. Similarly, other transducers with higher effective impedance would demand higher input impedances of the following amplifier.

Problems of another sort will show up if the value of R_f is greater than 2 megohms, or so. This is especially true if operational amplifiers of the low cost 741 family are used. In general, then:

1. R_{in} should be at least 10 times the effective impedance of the driving source.
2. R_f should be less than a couple of megohms unless high grade operational amplifiers are used.

A further rule that is not strictly necessary, but seems to make things a little easier, is to avoid trying to get too much gain out of a single stage. Try to limit the gain in any single stage to some figure in the "under 200" range. If greater gain is required, go to an additional stage. Operational amplifiers are cheap, in most cases, so it can be penny-wise and pound-foolish to try to get all your gain out of one.

Noninverting Followers

The noninverting operational amplifier follower (see figure 4) provides an output that is in phase with the input signal. It also provides a much higher input impedance.

The input impedance of an inverting follower is limited by the value of input resistor R_{in}. The noninverting follower, on the other hand, has an extremely high input impedance. If a premium operational amplifier integrated circuit is selected, then it may well be over 1 megohm. This is high enough to be useful with almost any imaginable driving source with a voltage output. Only certain high output impedance current transducer sources would require higher amplifier input impedance.

Differential Amplifiers

A differential amplifier will produce an output voltage proportional to the differ-

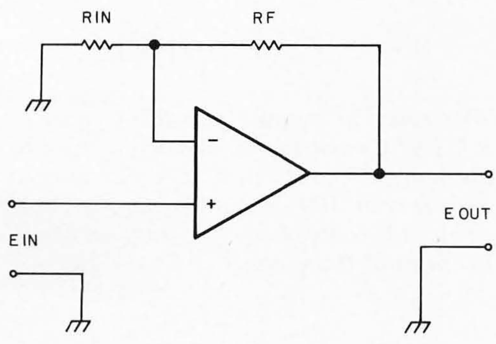


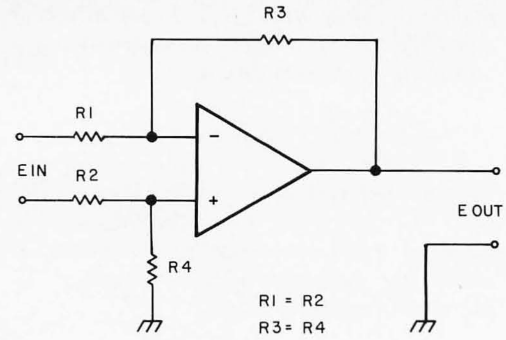
Figure 4: A noninverting amplifier. With the input signal applied to the noninverting (+) terminal, the output polarity will be the same as the input. The output voltage can be taken as the sum of the voltage drops across R_{in} and R_f if the voltage across the operational amplifier V_{ac} is taken as zero. Thus, the voltage gain A follows the relationship:

$$A = \frac{E_{out}}{E_{in}} = \frac{R_f + R_{in}}{R_{in}}$$

Figure 5: This is the basic diagram for a differential input amplifier. It is basically a combination of inverting and noninverting amplifiers. The output voltage will be:

$$E_{out} = \frac{R_3}{R_1}(E_1 - E_2)$$

This is provided that $R_1 = R_2$ and $R_3 = R_4$. Because this circuit tends to reject signals common to both input lines it is used to minimize interference picked up in wires connecting the transducer and its electronics.



ence between the two potentials referenced to ground as seen by the inverting and noninverting inputs. Almost all commercial operational amplifiers can also be used as differential amplifiers.

Figure 5 shows one type of differential amplifier that offers the economy of requiring but one operational amplifier.

This amplifier will serve nicely where compactness is required and both moderate gains (less than 500) and relatively low input impedance can be tolerated. Even the lowly 741 will serve nicely in this circuit. Higher gains and higher input impedances are possible with higher grade operational amplifiers such as the 725.

Another approach to higher gain and much higher input impedance which still uses operational amplifiers of moderate cost is the instrumentation amplifier of figure 6.

Operational amplifiers A1 to A3 can be 741 devices in many cases, but it is considered wise to use premium quality amplifiers for A1 and A2 in even moderately critical applications. The 1456 is a good choice in these cases, while the 725 or several equivalents are best in tight cases.

Resistor R8 is a multiple turn potentiometer used as the common mode rejection control. The ideal differential amplifier will only respond to a differential voltage between the two inputs. Unbalances and errors in real operational amplifier circuits, though, make it possible for the amplifier to respond to a common-mode voltage which is shared by both inputs. R8 is adjusted to null the output voltage under conditions where both inputs are tied together and a signal of several volts is applied to their common junction.

Some Comments on Practical Circuits

Any of the circuits shown in this article may be made workable by almost any builder with even moderate electronic assembly skills. It is only necessary to follow the rules that are given. It must be noted, though, that a lot more is possible with operational amplifiers, but would require somewhat greater design acumen. The rules can be violated by the sophisticated designer because they are selected to be the "most true for the biggest number." Next month, I'll continue with a discussion of some practical applications of operational amplifiers.■

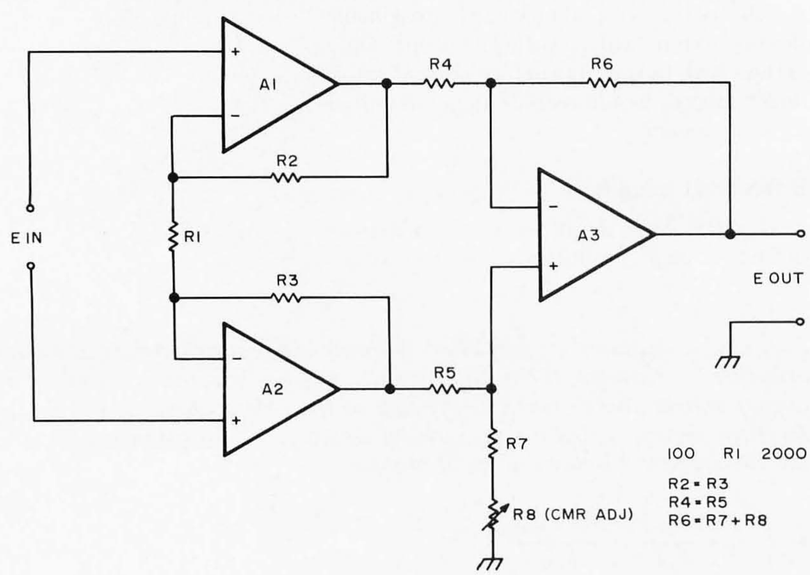
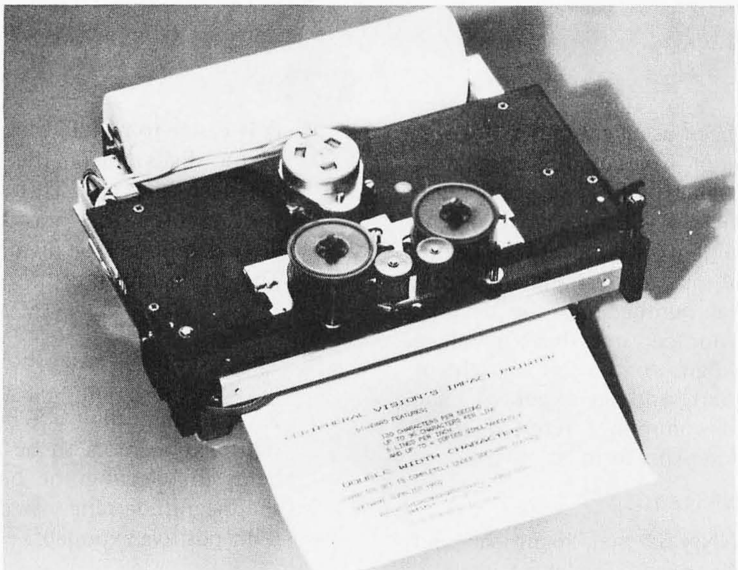


Figure 6: This circuit can be described as combining the simple differential amplifier of figure 5 and two copies of the noninverting gain follower of figure 4. The voltage gain A is given by:

$$A = 1 + \left(\frac{2 \times R2}{R1}\right) \times \left(\frac{R6}{R4}\right)$$

This equation assumes that R2 is equal to R3, R4 is equal to R5, and R6 is equal to the sum (R7 + R8). Also, it is wise to keep R1 between 100 and 2000 ohms in most cases with contemporary operational amplifier integrated circuits.

Peripheral Vision impacts your computer.



WITH A FULL-SIZE, LOW-COST IMPACT PRINTER.

Until now, the hobbyist and small businessman have had one major problem in assembling a reasonably price microprocessor system with the capabilities found in the more costly computers. It was impossible to find a high-quality, high-output printer for hard copy needs at an affordable price.

Peripheral Vision has come up with a solution.

We are offering a full-size *impact* printer designed for microprocessors—and it comes with a mini price. Kit prices start as low as $495 for the printer and interface card. And that won't impact your pocketbook.

Peripheral Vision's printer is loaded with capabilities. Take a look:

- It's fast—120 characters per second
- 96 characters per line, 12 characters per inch horizontal, 6 lines per inch
- Makes up to 4 copies simultaneously
- Character set and pitch variable under software control
- 5 x 7 character matrix
- Ribbon has built-in re-inkers for a life of 10,000,000 characters
- Paper can be either a standard 8½-inch roll, fanfold or cut page
- Interfaces to 8-bit parallel ports

Just remember, Peripheral Vision is committed to helping you get along with your computer. The new printer we are offering is another example. It is high quality, low in cost and will definitely impact your system.

Write or call now to find out how to impact your computer.

PERIPHERAL VISION INC

P.O. Box 6267 Denver, Colorado 80206 (303) 777-4292

What's in a Floating Point Package?

Sheldon Linker
3514 Boelter Hall
UCLA
Los Angeles CA 90024

If you have been using computers for any length of time, or have used a calculator, you know the value of floating point numbers. In this article, I will endeavor to show how floating point works and how to use it.

Initially, you must understand the representation of real numbers. In the decimal system, a real number may have a sign, a decimal point, digits to the left or right of the decimal point, and an exponent. The general form is commonly referred to as scientific notation. This form is:

$$\pm N.NNNNNNX10^{\pm MM}$$

where N.NNNNNN is the "mantissa" and MM is the "exponent." Unfortunately, decimal numbers are hard to deal with, even with the decimal add or adjust instructions of some microprocessors. There are some simplifications that can be made without serious loss of precision:

1. It is easier to handle the numbers if all of the digits are on the same side of the decimal point. This simplifies the shifting used to "normalize" results after a multiply or divide instruction. The decimal form of this is:

   $$\pm .NNNNNNNX10^{\pm MM}$$

2. The exponent base should, for simplicity of programming, be a power of 2. Assuming that a 6 bit (excluding sign) exponent is to be used, table 1 helps in selecting the base by giving the magnitude range (in decimal) for a 6 bit positive exponent.

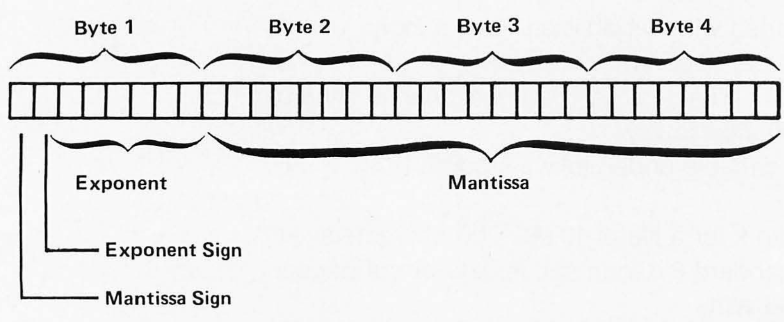
For the sake of conformity to many existing floating point packages, you should either use a base of 2 or 16. A base of 2 gives somewhat limited dynamic range; so for the rest of the article, I will use base 16. The dynamic range is defined as the difference between the high decimal value of the exponent and the low value. This reduces the generalized form to a more specific case.

$$\pm .NNNNNNNX16^{\pm MM}$$

Another consideration is storage. So far, I have 6 bits allocated to the exponent, 1 bit for the sign of the exponent, 1 bit for the sign of the mantissa, and some amount for the mantissa itself. As one byte has already been used, another three would work out nicely. 24 bits gives a precision of 24Xlog2 or 7.22 digits of accuracy (which is treated as 7 for formatting purposes).

Taking all of the above into account, we now have the real number stored as in figure 1. *[This 4 byte format is similar to the format used in an IBM 360 or 370 for single precision floating point.]*

We now come to what you've been waiting for: arithmetic. Addition and subtraction are the easiest, so I'll start with them. In the algorithms I'll present, addition and subtraction are simplest when using only



Example: The value of hexadecimal 013243F7
is hexadecimal $0.3243F7 \times 16^1$
or hexadecimal 3.243F7
or decimal 3.141593

Figure 1: Details how the floating point number may be stored in memory. In this format the first byte contains the exponent, the sign for the exponent, and the sign for the mantissa. Bytes two, three and four contain the mantissa of the number in absolute value form, with 24 bits of precision.

positive terms; so the substitutions shown in table 2 can be made. Vertical bars denote absolute value as in |X| signifying the absolute value of X.

Table 2 actually boils down to a series of additions, subtractions and sign inversions. A flowchart of this process is given in figure 2.

When adding or subtracting, the first step is to determine whether a number is negative. When the first byte of a floating point number is negative, then the entire number is negative.

The second step is to change the sign of a number. This is done by adding or exclusive orring a hexadecimal 80 to the first byte of the number in this format.

You may notice that in the flowcharts I took no action to take the absolute value of either the A or B terms. This is because the mantissa bytes are actually in absolute value form.

You are now ready for the next step in addition or subtraction. Before the numbers can be used together, the exponents must match. This is done by incrementing the smaller exponent until it equals the larger exponent, while at the same time dividing the corresponding mantissa by 16. The larger exponent becomes the exponent of the answer up to this point. When comparing

Base	Maximum Value (Base raised to 64th power)
2	1.84×10^{19}
4	3.40×10^{38}
8	6.28×10^{57}
16	1.16×10^{77}
32	2.14×10^{96}

Table 1: A summary of the dynamic ranges, difference between the lowest and highest expressable numbers, of several different bases. The bases are all powers of two since these are the easiest to handle on a computer. The range of the number is determined by assuming an exponent of 6 bits, excluding the sign.

Original Equation	Use
A+B A⩾0 B⩾0	A+B
A+B A⩾0 B<0	A − |B|
A+B A<0 B⩾0	B − |A|
A+B A<0 B<0	−(|A| + |B|)
A−B A⩾0 B⩾0	A−B
A−B A⩾0 B<0	A + |B|
A−B A<0 B⩾0	−(|A| + B)
A−B A<0 B<0	−(|B| − |A|)

Table 2: Substitutions that may be made when adding or subtracting. This table reduces to a series of additions, subtractions and inversions which are performed by the routine of figure 2.

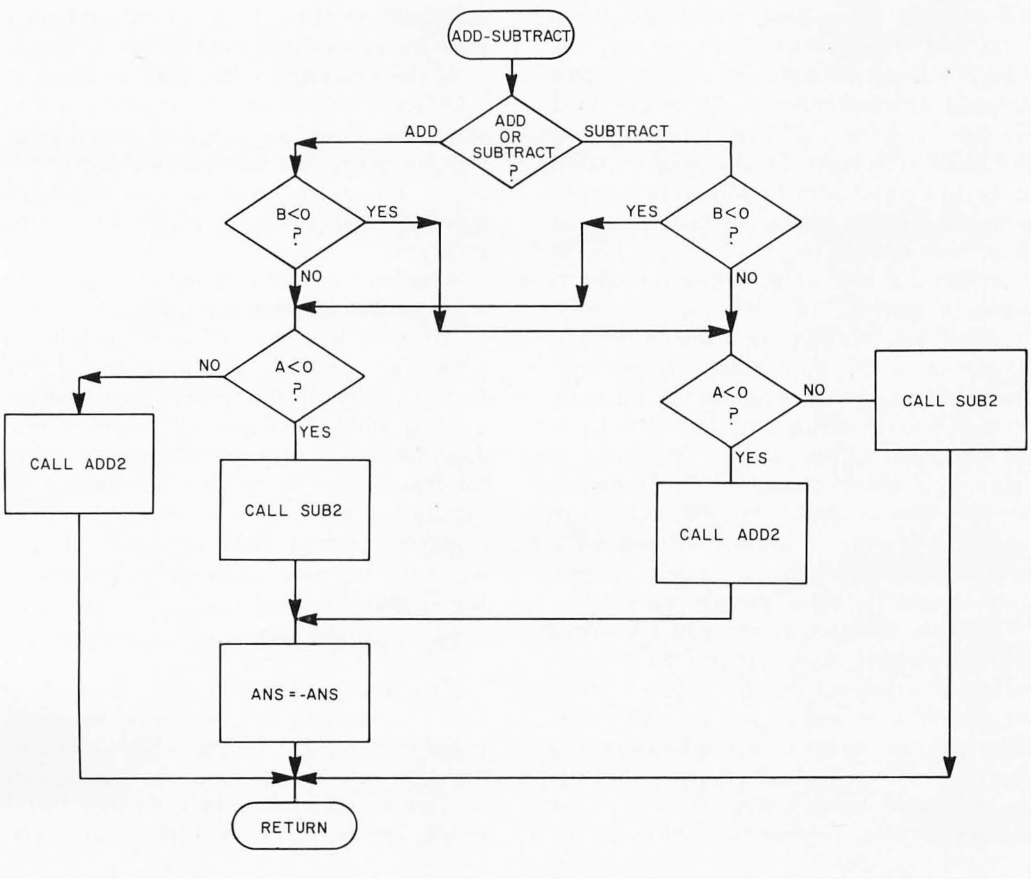


Figure 2: Flowchart for a routine which will manipulate the two values to be added or subtracted according to the rules summarized in table 2.

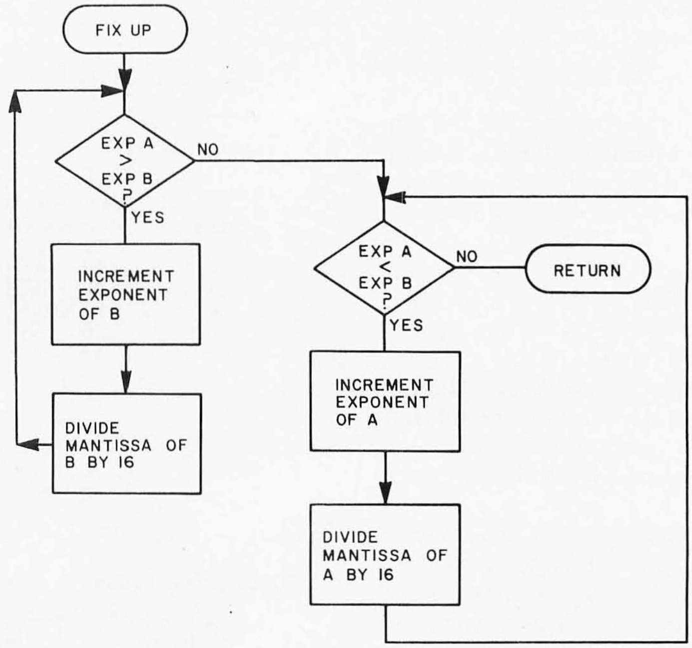


Figure 3: The FIXUP routine manipulates the exponents of the two numbers so that they are equal. This is a prime consideration for subtraction and addition routines.

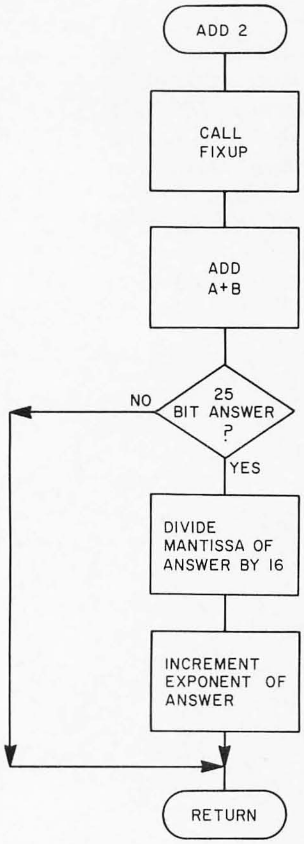


Figure 4: This addition routine adds the two numbers and then checks if there has been an overflow into bit 25. If this condition exists, the mantissa is divided by decimal 16 and the exponent is incremented by one, thus giving a 24 bit answer.

one exponent to another, you must keep in mind that the exponent occupies only the right 7 bits of the byte. To convert it to a standard signed number, you have to extend the sign by doing a shift left followed by an arithmetic shift right. Finally, you are about ready to do the actual addition or subtraction, but so that nothing is lost, you must treat the mantissas as 32 bit fields. A flowchart for the whole alignment step is shown in figure 3.

After the addition or subtraction you may have a 25 bit number, taking into consideration the carry of the 24 bit mantissas. If you do, divide it by 16 and add 1 to the exponent of the answer, as shown in figure 4. If the exponent of the answer is now 64, then you have the condition known as overflow. This is usually considered a severe error and warrants stopping the program, or substituting a "default" value.

After the subtract, you may be left with a negative number or zero. If you are left with a negative value, set the sign indicator bit of the answer to 1 and change the sign of the difference, as shown in figure 5. There is a final step shown in figure 6, after the subtraction called normalization. If the mantissa of the answer is less than hexadecimal 0.100000, multiply it by 16 and subtract 1 from the exponent of the answer.

If the exponent is less than or equal to −64, you have a condition known as *underflow*. The underflow condition is considered a minor error, and the accepted response is to set the answer to zero, print a warning message, and continue execution of the program.

Continue normalizing until the mantissa is at least hexadecimal 0.100000.

To avoid the error of coming up with an answer of −0, apply this simple test: If the first 16 bits of the answer equals hexadecimal 8000, then the real answer is zero. Also, be sure to assemble the exponent into the final answer along with the sign and the mantissa.

After mastering addition and subtraction, we now forge onward to multiplication. As theory tells us:

$$(A \times 16^B)\ (C \times 16^D) = AC \times 16^{B+D}.$$

This makes things extremely simple. Step one is to multiply the two 24 bit mantissas together, giving a 48 bit result. Don't scream! You probably already have a method of multiplying two 8 bit numbers, giving a 16 bit result. The method of multiplying and sum-

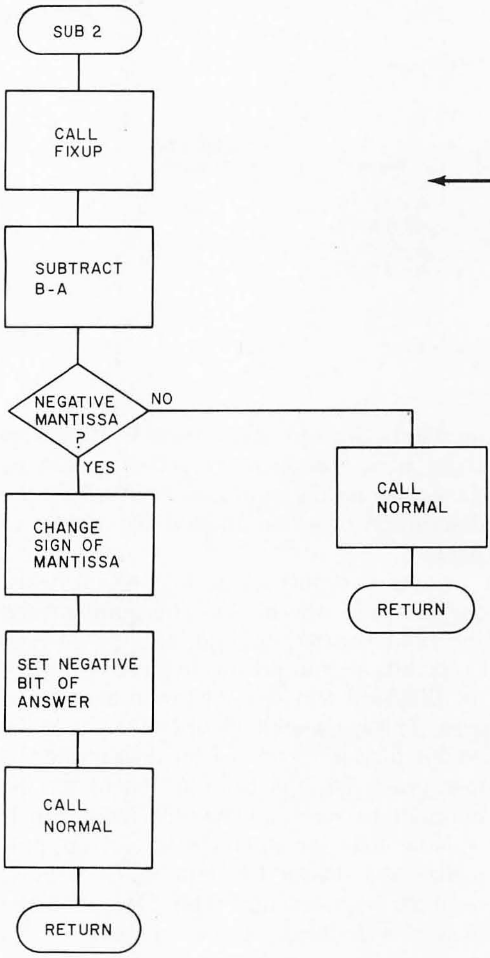


Figure 5: This subtraction routine subtracts two numbers and then determines the sign of the answer. If the sign is positive the answer is normalized and the number is then returned to the calling program. If the answer is negative a positive number is sent to the normalization routine and then the sign of the number is set negative before being returned to the calling routine.

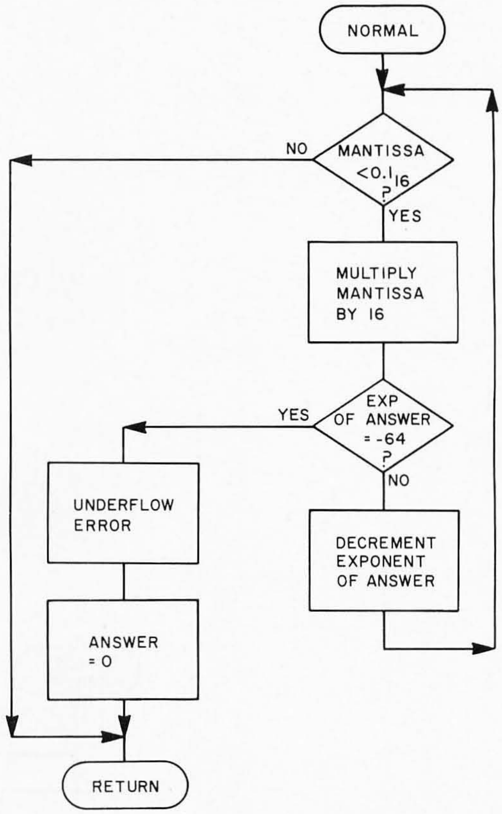


Figure 6: This is a normalization routine for the floating point format used in this article. The answer is set up so that the mantissa is hexadecimal 0.100000 to hexadecimal .FFFFFF. This is accomplished by multiplying the mantissa by decimal 16 and then incrementing the exponent until the high nybble of the mantissa is from hexadecimal 1 to 16.

ming for a 48 bit answer is shown in figure 7. After carrying out the nine 8 by 8 multiplies shown, sum vertically. Throw away bytes K and L of the result since they won't show up in the answer. Step two is to normalize bytes G, H, I and J. Bytes G, H and I now become the mantissa of the answer. Step three is to find the sign of the answer. Since table 3 applies to both multiplication and division, the exclusive or function can be used on bytes A and B to find the sign of the answer. Step four is to find the exponent. Separate the exponents, as in the FIXUP routine flowcharted in figure 3, and add them together along with the exponent given by the normalization process. Check for overflow and underflow.

The rule for division is stated as:

$$(AX16^{B}) \div (CX16^{D}) = A \div DX16^{B-D}.$$

This is similar to multiplication, except that the exponents are subtracted, along with the difference that you divide the mantissas instead of multiplying. To multiply, you took two 24 bit numbers and got a 48 bit product. To divide, you take a 48 bit dividend and a 24 bit divisor, giving a 48 bit

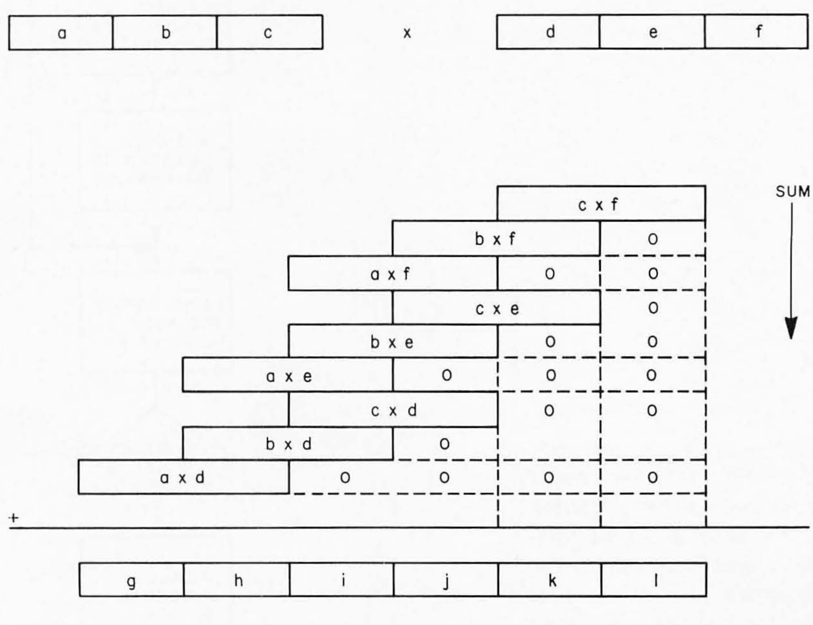


Figure 7: A summary of the method of multiplying two 24 bit mantissas to attain a 48 bit mantissa result. After the summation, the low order K and L bytes may be discarded as they will not show up in the final answer.

Table 3: A summary of the sign of the answer given the sign of the numbers used in a multiplication or division.

Signs	Sign of Answer
A≥0 B≥0	+
A≥0 B<0	–
A<0 B≥0	–
A<0 B<0	+

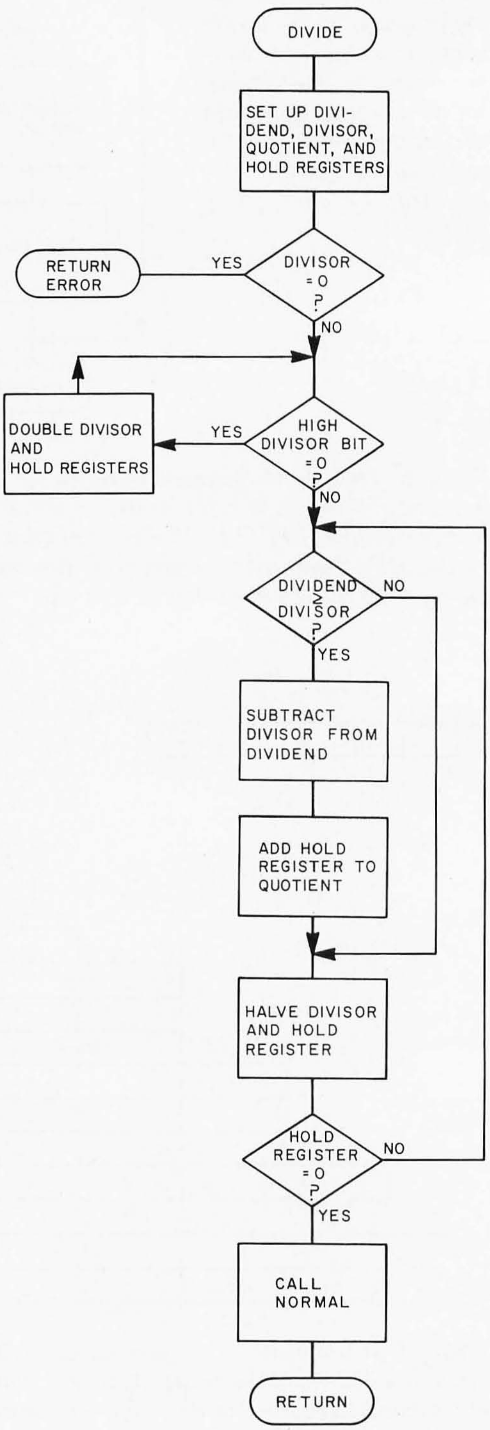


Figure 8: A division routine for floating point numbers. This routine starts with a 48 bit dividend and a 24 bit divisor to arrive at a 48 bit quotient. The routine then normalizes the answer before returning it to the calling program.

quotient. Unfortunately, there is no way to chain partial quotients together as can be done with partial products. As you read the description of 48 bit division, please refer to figure 8.

First, you must set up four 48 bit fields: the dividend, the divisor, the quotient and the hold register. Second, set the 24 high order bits of the 48 bit dividend to the 24 bit dividend and the 24 low order bits to zero. Third, set the high order 24 bits of the 48 bit divisor to the 24 bit divisor and the low order 24 bits to zero. Then, set the quotient to zero and the hold register to 1.

Now that the numbers are set up, normalize the divisor by multiplying it by 2 until the high order bit is on. Every time the divisor is doubled, the hold register is also doubled. Notice that if the divisor is zero, it will never get normalized; so it's a good idea to indicate an error condition and abort the program if this is the case.

The next step is the repeated subtraction: If the dividend is greater than or equal to the divisor then subtract the divisor from the dividend and add the hold register to the quotient. Then divide the divisor and the hold register by 2. If the hold register is still nonzero, then keep performing the repeated subtraction. This leaves only the normalization and sign operations, as in multiplication.

This completes the summary of basic floating point operations needed to do calculations in scientific notation. There are other formats which can be used, with more or less precision, using BCD coding for the mantissa and exponent, etc. The general steps required are all very similar. Other items which are useful to have, but which I have not covered here, include conversion routines for input and output, and transcendental function evaluations. But even if you can't calculate a sine or a cosine, and don't have a flexible FORTRAN style formatting output, the basic calculations of addition, subtraction, multiplication and division can prove a useful adjunct to your software.■

WE HAVE IT. THEY DON'T.

SOFTWARE

DOS

Disk Operating System *Device management and executive scheduling *Directoried file management system *Sequential and indexed random file structure *Full support of all languages, utilities, and peripherals.

FORTRAN IV

ANSI FORTRAN compiler *Double precision real and complex arithmetic *Non zero based arrays *reentrant subroutines *Random file I/O *ISA process control, real time, and multitasking extensions *Chaining, swapping, and overlays *Commercial subroutine package.

MACRO ASSEMBLER

Relocatable code generation *Macros *Dynamic instruction definitions *Conditional assembly.

EDITOR

Character or line oriented *String operations for search, search and replace *Store and execute command strings *Multiple files open concurrently.

HARDWARE

CPU

Micro NOVA® 16 bit processor *Real time clock *Hardware multiply and divide *Power fail auto restart *8K bytes RAM *9 slot chassis with front panel and power supply. (optional 18 slot chassis)

ASYNCHRONOUS SERIAL INTERFACE

Strap selectable data, parity, and stop bits *Modem control signals *Data rate strappable from 50 to 19,200 baud *EIA-RS232C or 20ma interface.

DEBUG ROM

Examine, deposit, control of memory and internal CPU registers and status *Breakpoints for monitoring running programs *Auto program load.

DISK

2 diskette drives per controller *315K bytes per diskette *31.5K bytes per second transfer rate.

EXPANSION

Memory expansion in 8K and 16K byte increments up to 64K bytes *General purpose interfaces for custom applications *9 slot I/O expansion chassis.

DISK SYSTEM

*CPU with 40K bytes RAM *Dual diskette *Asynchronous serial interface *CRT terminal *Debug ROM *DOS *FORTRAN IV *Macro Assembler *Editor *Utilities and support programs *Complete documentation set. **$8400**

Other systems available from **$2700**
All hardware is fully assembled and tested. Support is available from service centers around the country. Prices and specifications subject to change without notice.

BPI

If you are serious about computing, call or write today. BPI Inc., 2205 East Broadway, Suite 6, Tucson, Arizona 85719 (602) 326-6975

® NOVA is a registered trademark of Data General Corporation, Southboro, Massachusetts.

Technical Forum

The 8080 High Level Language Project of Peter Skye, Continued

Peter Skye, Chief Engineer
Watermark Inc
10700 Ventura Blvd
N Hollywood CA 91604
(213) 980-9490

In the August issue (see page 12), you published my letter about writing a high level language compiler for use on a microcomputer. I got quite a bit of response to this project, and I'd like to fill your readers in on what is going on.

First Item

The following languages were mentioned as possibilities for a new compiler, with PL/I the overwhelming favorite. In order of preference: PL/I, FORTRAN, BASIC, SNOBOL, PASCAL, COBOL, RPG, FORMAC, TRAC, C, APL. ALGOL was not mentioned.

I chose an expanded PL/I (which I am exuberantly and unabashedly calling PL-SKYE) for the following reasons:

1. PL/I includes all of FORTRAN, BASIC, and ALGOL.
2. The main attraction of SNOBOL is a few string handling functions. These have been incorporated into PL-SKYE.
3. PL/I includes all of COBOL, although the format and syntax of COBOL is different from PL/I.
4. RPG is a very simple language. RPG programs are very easily rewritten in PL/I.
5. Only one person mentioned APL. Since APL's "thing" is the power in its assignment statement, I have included the APL operators in PL-SKYE. Question: APL operators have no hierarchy, whereas PL/I operators do. What hierarchy should be implemented? (I have made all the APL operators the same.) Come on, APL fans, *write* me this time!
6. Format ability has been included in PL-SKYE.
7. I am unfamiliar with PASCAL, TRAC, and C, but the few people who mentioned these languages didn't mention anything that PL/I couldn't do.

Second Item

See figure 1 for the proposed character set for PL-SKYE. This character set is larger than those in common use, although nothing out of the ordinary for a textbook. Which brings up some questions:

1. Are there any symbols that I have forgotten?

Figure 1: Character set, in logical groupings. Where more than one version of a symbol is in common usage, all are shown.

1) blank, 0-9, A-Z, a-z
2) Greek alphabet (upper and lower case). Must include
   α, ε, ρ, ω } for APL
   Σ summation symbol
   Π multiplication symbol
   π pi
3) . , : ; ! ? ( ) { } [ ]
4) +
   —
   - APL negative symbol
   ****
   x
   ÷
   /
   \
   |
   & &
   ∧
   ∨
   <
   >
   ≤
   ≥
   =
   ≡
   ¬
   ~
5) ⩚ combinatory symbol

APL symbols
6) ≠ ≀
   ⌿ ⌈
   ⍀ ⌊
   ⊤ □
   ⊥ ⌹
   ⌶ ○
   ⍎ ⌽
   ⍕ ⍉
   • ⊖
   •• ⊛ ⊛ ⊗
   ⊂ ∇
   ⊃ Δ
   ∩ ⍒
   ∪ ⍋
7) $ $ $
   ¢ ¢
   %
   #
   ♭ musical flat symbol
   @
   '
   "
   ° degrees symbol
   ←
   ↑
   →
   ↓
   ■
   ∴ therefore symbol
   ||
   ∫ integral symbol

A STORY IN THREE ACTS

ACT 1: Our story begins with the ACT-I terminal—the world's first Affordable Computer Terminal. The ACT-I is an exceptionally practical interface between man and machine for any computer system which communicates over a serial ASCII link. Whether querying a microprocessor based system or time-sharing on a major computer net, the ACT-I is the most economical method of alphanumeric communication at data rates from 110 baud all the way to 9600 baud. The ACT-I video computer terminal manages a 1024 character display memory organized as 16 lines of 64 characters chosen from the standard upper case ASCII set. Receipt of more than 64 characters on a line or the 'CR' code initiates a scroll operation. The entire screen may be locally cleared at any time by depressing the 'CLEAR' key.

I/O SPECIFICATIONS
(JUMPER SELECTABLE):

DATA RATE: 110, 300, 600, 1200, 2400, 4800, 9500 BAUD

PARITY: odd, even, or none

STOP BITS: 1 or 2

LOGIC LEVELS: RS232, TTL, or 20 ma loop

PRICE: $400





ACT 2: The plot thickens. For remote communication to a central processor MICRO-TERM introduces the ACT-II. The ACT-II includes all of the desirable features of the ACT-I with the important addition of an integral originate—only 300 baud modem and acoustic coupler for a standard telephone handset. The ACT-II (without monitor) slips easily into a briefcase (4x14x11) and readily commutes with you.

The ACT-II's modem features 5 stages of active filtering as well as one passive stage to achieve the lowest possible error rates despite the noises present on Ma Bell's lines.

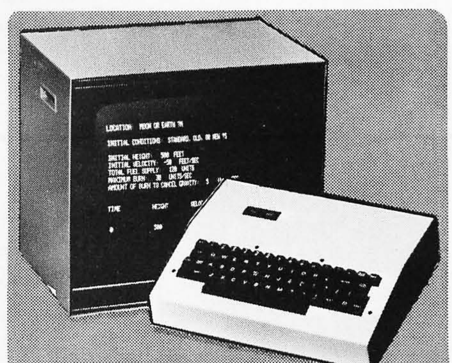
Want true CPU power for low cost? Investigate the time-sharing systems in your area; the ACT-II can be your key to the tremendous libraries of debugged, documented, software which are available on these dial-up systems.

PRICE: $550.

ACT 3: Not to be upstaged by its exciting predecessors, the ACT-III performs with state of the art elegance and versatility. The ACT-III includes switch selectable display formats: 24 lines of 80 characters; 48 lines of 40 characters; or 96 lines of 20 characters. Transmission is also switch selectable between three modes: character, line or page at a time mode. The PRINT key sends the contents of the 1920 character display memory out an RS232C printer port at rates up to 9600 baud. For optimum efficiency, trailing spaces on any display line are excluded from transmission and of course the printer rate is independently switch selectable.

Full cursor control, direct cursor addressing, protected data fields, and video inversion are all standard encodeable functions. Convenience features such as a video monitor AC power outlet and a switch to initiate a comprehensive terminal self test are available on the cabinet rear. The ACT-III is the most complete video terminal available at a price far below that of the competition, smart and dumb alike.

PRICE: $700.



GENERAL INFORMATION

All MICRO-TERM products are fully assembled, tested and guaranteed for 90 days.

The MICRO-TERM product line is available from stock at discriminating computer stores or may be purchased directly from the factory (30-45 day delivery). Prices are F.O.B. St. Louis, Mo. excluding video monitors. Optional high resolution video monitors are available from the factory beginning at $125.00.

BankAmericard and Master Charge accepted.

MICRO-TERM INC.
P.O. BOX 9387
ST. LOUIS, MO 63117

2. Is there any standard for bit representations on these characters (IBMers, please check your "green cards" 'cause I lost mine)? [*Use ASCII . . .* CH]
3. How hard will it be to implement these characters on 5 by 7 dot matrix displays and printers?

Again, I'd appreciate your responses, especially from manufacturers concerning #3.

Third Item (Machine Language, Pie in the Skye)

One of the most common requests was for the ability to have in line "assembly language" statements. I have made the value judgement not to do this, for the following reasons:

1. All of the examples of things "you couldn't do" really can be done in PL/I, by using the built-in functions, iSUB capability, and list processing. And programming is faster this way, because you get accurate diagnostics.
2. In a properly designed high level language, there should be a function or keyword to do the thing you're trying to accomplish, without having to go to assembly language. PL-SKYE will be updated, just like the dictionary, so that it will indeed do everything. (And it *isn't* "too big!") [*Good luck . . .* CH]
3. If you put in statements which restrict you to a certain processor, your programs are not going to be compatible on any other system. Listen, folks, processor independence is where it's at. PL-SKYE will be written completely in PL-SKYE. I had to add a few extra keywords to the language to accomplish this, but that's what you're *supposed* to do when you write a better compiler.

Fourth Item

Relocatable code was mentioned several times. Yes, PL-SKYE has this. Nobody mentioned it, but relocatable code makes it easier to get into a multiprogramming environment (play Star Trek, Pong, *and* cook supper all at once!?), and allows concepts of modular programming and linkage editing not possible otherwise.

Fifth Item

As to which processor to write for, people mentioned the 6800, 8080, Z-80, and Micronova. Although my original intentions were for a Z-80 system with floppy disk, I now plan on having code for the other processors as well. And, you won't *have* to have disk.

Sixth Item

Many people want the compiler's and program's IO to be independent of some particular peripheral device. It sounds like they really want the IBM Operating System, which I'm not opposed to at all. But they'll have to wait; that kind of code doesn't belong in a compiler. You'll have to compile your IO either directly to a peripheral, or through an IBM type of Data Control Block (DCB). How about somebody else writing a good OS (Operating System)? In PL-SKYE?

Seventh Item

Status of PL-SKYE? The original (pre-BYTE letter) version has had two major modifications: The input token routine has been expanded to accept APL operators, and the code generation logic has been expanded to handle DECLARE statements internal to contained blocks. I think the easiest way to distribute PL-SKYE will be on a phonograph record, which you will play through your RS 232. Any comments on this idea? [*Why not bar codes?. . .* CH] ■

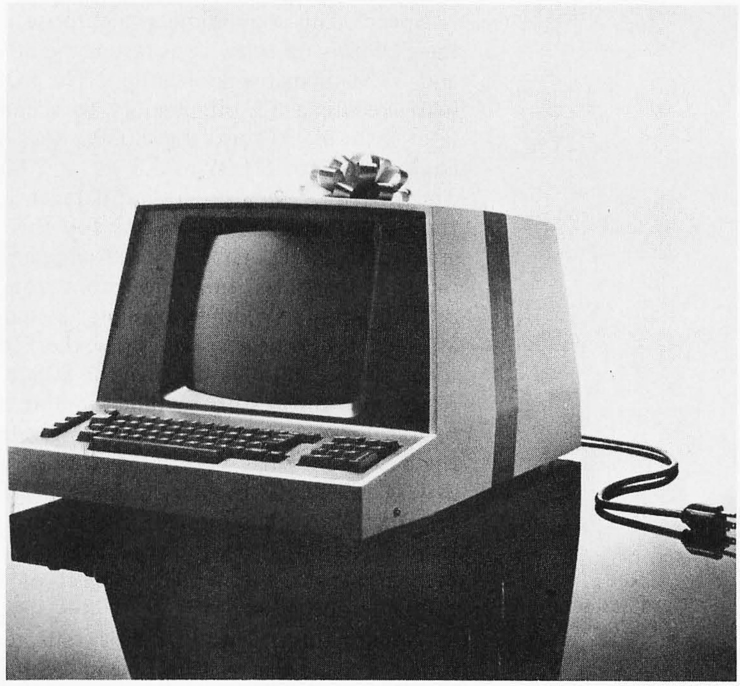
More Comments on PDP-8s (Or, "Over the Hill" Means "Worth Looking At")

I write this in response to the letters of Mr Simmons and Mr Gordon, regarding the viability of the PDP-8 as a computer for the home hacker [*"A Salute to the PDP-8 and Farewell" BYTE #14, page 138; "Comments on Design Evolution" BYTE #16, page 50*]. In evaluating any computer for possible use, several factors are always present, such as price, size, power consumption, reliability, software and so forth. Every user will have different criteria; I would like to offer some comparisons, based on a set of criteria that might be of interest to the home user, that show the PDP-8 in a much more favorable light than either Mr Simmons or Mr Gordon show.

Price: PDP-8s, 8/Is, 8/Ls are now available on the used computer market for $750 to $1200. These are for assembled, stand alone units with 4 K of core memory, Teletype interface, etc. The PCM-12 is available, as a kit, for about $1000 for the same equipment (although with semiconductor memory). These prices are very close to

XIMEDIA PRESENTS

The SOROC IQ120



CURSOR CONTROL. Forespace, back-space, up, down, new line, return, home, tab, PLUS ABSOLUTE CURSOR ADDRESSING.

TRANSMISSION MODES. Conversation (half and full Duplex) PLUS BLOCK MODE — transmit a page at a time.

FIELD PROTECTION. Any part of the display can be "protected" to prevent overtyping. Protected fields are displayed at reduced intensity.

EDITING. Clear screen, typeover, absolute cursor addressing, erase to end of page, erase to end of line, erase to end of field.

DISPLAY FORMAT. 24 lines by 80 characters (1,920 characters).

CHARACTER SET. 96 characters total. Upper and lower case ASCII.

KEYBOARD. 73 keys including numeric key pad.

REPEAT KEY. 15 cps repeat action.

DATA RATES. Thumbwheel selectable from 75 to 19,200 baud.

SCREEN. 12 inch rectangular CRT — P4 phosphor.

SPECIAL INTRODUCTORY PRICING

Kit $ 995.00 **Assembled $ 1,295.00**

(Price includes block mode, lower case and 24 line options.)

IMSAI + Z-80

XIMEDIA is offering a special price on an IMSAI I-8080 with a TDL ZPU Z-80 board instead of the standard 8080 cpu.

The system includes:

IMSAI I-8080 without cpu board. 22 slot mother board with 22 *pre-soldered* connectors and card guides; cooling fan.

Price:

Kit	Assembled
$ 999.00	$ 1,295.00

Selectric Terminals $1,200.00

AT LAST!

An ASCII coded Selectric terminal with a RS232-C interface.

Call or write us today for details.

XIMEDIA OFFERS A FULL RANGE OF PRODUCTS FOR THE PERSONAL COMPUTER ENTHUSIAST AND THE SMALL SYSTEM DESIGNER. LET US QUOTE ON ALL YOUR HARDWARE AND SOFTWARE NEEDS.

OUR RETAIL STORE — THE COMPUTERIST™ — IS NOW OPEN IN SAN FRANCISCO. CALL US FOR DIRECTIONS.

XIMEDIA

1290 24th Avenue • San Francisco, CA 94122

(415) 566-7472. Please call collect!

California residents add 6% sales tax. Please allow 3 weeks for delivery.

those of unassembled Altair 8800 or IMSAI 8080 machines with similar equipment.

Software: No contest – the PDP-8 has more software available than any other machine in the mini or microcomputer class: two disk monitor systems, a real time system with batch mode, four assemblers, three editors, a variety of FORTRAN systems, DIBOL, FOCAL, several BASIC systems, and numerous special purpose packages.

Speed: This may come as a surprise, but the PDP-8 is up to twice as fast as the 8080a and 6800 in many applications. The PDP-8 will execute a 12 bit memory to accumulator add in 3.0 μs; the 8080a does the equivalent with 8 bits in 3.5 μs, at 2 MHz. The PDP-8 is thus processing data at 1.75 times the rate of the 8080a for this restricted example. Granted, the comparison of two machines for throughput rate is difficult, but any other example, including assorted test routines, will show the PDP-8 to be the equal or better than the 8080a, or other microprocessors. While the IM6100, operating at 4 MHz, is slightly slower than the 8080a, the high speed IM6100A operates as fast as a PDP-8/E, giving a speed of 2.5 μs for the above mentioned instruction example. The IM6100A thus operates at a rate comparable to the 4 MHz version of the Z-80 microprocessor.

Architecture: The PDP-8 is repeatedly called a primitive machine, and both Mr Simmons and Mr Gordon make much of comparing it to the 8080a, among others. But consider these facts: The PDP-8 has true indirect addressing; the 8080a and 6800 do not. The "cute" auto index registers, contrary to Mr Simmons' feelings, are quite powerful, and are used in virtually all software for the PDP-8 to good advantage. The statement that " . . . no program is really quite sure they are not used by some other program . . . " makes little sense. The microprogrammed instructions, while confusing at first, allow the PDP-8 to do some operations in one instruction that take several in a 8080a or 6800. For example, the skip instructions allow testing up to three conditions at once, something which would take three jump instructions in a 8080a. The PDP-8 and 8/I have optional hardware multiply/divide. The IM6100, while not available with this option, can implement a vectored priority interrupt structure with the PIE chips that are made for this device. The paging structure of the memory is a nuisance, and the subroutine linkage is inconvenient; however, since all PDP-8s have core memory, the fact that read only memory or programmable read only memory cannot be used is less of a problem than with the new microprocessors, which generally use semiconductor memory. Note that this is not true for an IM6100 based machine, and some form of external storage is required in this case.

Power consumption: This is a very real consideration, since most home hackers can ill afford the extra cost of running a power hungry machine. Here, the PDP-8 is at a definite loss; a PDP-8 uses about 500 W, an 8/I or 8/L about 300. This is over twice what a typical 8080a based system uses. However, the IM6100 is a CMOS device, and a machine based on it would use less power than other micro machines.

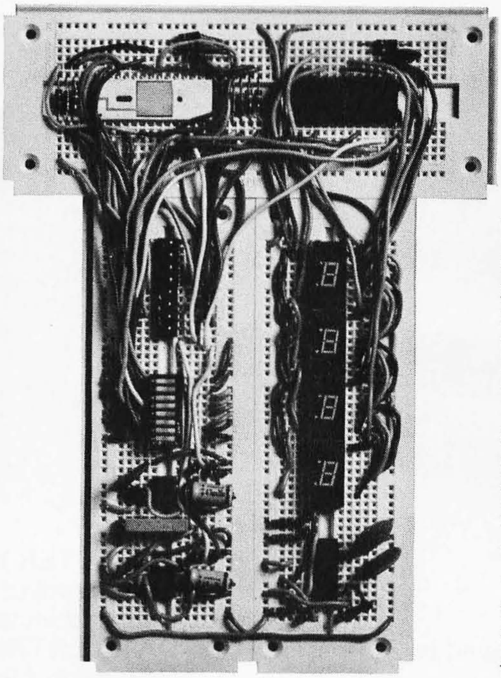
Interfacing: Here again, the PDP-8 is in second place. Without building a special interface, all the new Altair bus products are unavailable to a surplus PDP-8 user. On the other hand, numerous surplus devices are available for interfacing to the PDP-8.

Memory: A fully expanded PDP-8 memory holds 32,768 12 bit words. This is equivalent to 75% of a fully expanded 8080a or 6800 memory, in terms of raw storage. While not as much as the new machines' capacity, this is still quite a large amount, and adequate for nearly any application.

I would like to offer some final thoughts on word length and design obsolescence. In what holy book is it written that a computer must operate on multiples of 8 bits? The 12 bit word of the PDP-8 makes it more adept at multiple precision arithmetic; the 8080a and 6800 are more comfortable with character oriented data. To offer some perspective, the IBM 360/370 series works on 8 bit characters, while the CDC 6600/7600 uses 60 bit words for fast arithmetic. The IBM 1620 used 6 bit characters for storage; both Mr Simmons and Mr Gordon will find that, historically, word length has had little to do with the relative merit of a computer design. As far as design obsolescence goes, a design becomes obsolete when its price to performance ratio no longer competes well with other designs. It is interesting to note that Digital Equipment is selling more of those "old war-horse" PDP-8 computers today than ever before; obviously it is nicely meeting the needs of someone – a clear indication its price to performance is still quite competitive. I submit that, on the basis of all this, the PDP-8 and IM6100 are both worth considering by the home hacker for his or her system. This "over the hill" machine, as well as the IM6100, will probably be around, and quite useful at that, for several more years.

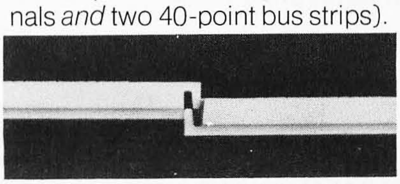
Ronald B Howes Jr
5753 Emerson Av S
Minneapolis MN 55419■

ANNOUNCING THREE NEW BREAKTHROUGHS IN SOLDERLESS BREADBOARDING.



Whether you design or build circuits for fun or for profit, you owe it to yourself to discover how fast and easy CSC solderless breadboarding can be. Now, more than ever. Because of three new breakthroughs in breadboard design. And our new EXPERIMENTOR™ sockets** that make the most of them.

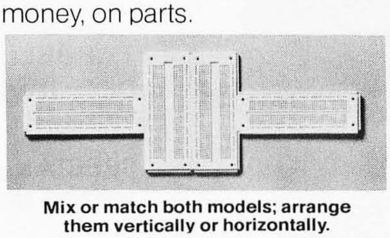
1. Price Who says a quality breadboard has to be expensive? For as little as $9.95, CSC's EXPERIMENTOR sockets let you design, assemble and modify circuits as fast as you can push in—or pull out—component leads. On a rugged one-piece socket with 550 solderless tie-points (94 five-point terminals *and* two 40-point bus strips).



Sockets lock together, snap apart to handle any size circuit with ease.

But don't let the low price fool you: EXPERIMENTOR sockets are precision-molded of durable, abrasion-resistant material, and feature CSC's non-corrosive, prestressed nickel-silver contacts for positive connections and longer life. All contacts are identified, too...with molded-in designations for easier circuit assembly and diagramming.

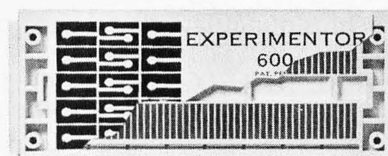
2. Compatibility CSC EXPERIMENTOR sockets end the "big-chip blues." They're the only ones with full fan-out capabilities for microprocessors and other larger DIP's, as well as 4-16-pin units. EXPERIMENTOR 600's 6/10" center is ideal for microprocessor's, clock chips, RAM's, ROM's, PROM's, etc. While EXPERIMENTOR 300's smaller 3/10" center is perfect for smaller DIP's. Both units, of course, accept transistors, LED's, resistors, capacitors, pot's—virtually all types of components with plug-in ease. As well as #22-30 solid hook-up wire for interconnections. Eliminating heat and lead damage to expensive components. And saving you more money, on parts.



Mix or match both models; arrange them vertically or horizontally.

3. Flexibility With CSC EXPERIMENTOR sockets, you can arrange your breadboard to suit your circuit...instead of vice versa. An exclusive snap-together interlocking system lets you instantly connect them. Vertically or horizontally. So you can mix or match 3/10 and 6/10" centers...expanding or contracting to meet your requirements.

CSC EXPERIMENTOR sockets can be used alone, or mounted on any convenient flat surface, thanks to molded-in mounting holes and vinyl insulation backing that prevents shorts. You can mount them with 4-40 flat-head screws from the front, or 6-32 self-tapping screws, from the rear. But however you use them, EXPERIMENTOR sockets are an unbeatable way to build and test twice the projects in half the time.



Both models feature 94 five-point terminals (vertical lines) and two 48-point bus strips (one strip shown).

WHY WAIT? CSC EXPERIMENTOR sockets are available now from your CSC dealer, or by phone from CSC, at $9.95* for the 300 and $10.95* for the 600. Call 203-624-3103 (East Coast) or 415-421-8872 (West Coast)—major credit cards are accepted.

*Manufacturer's suggested list price. Prices subject to change without notice.
U.S. Pat. No. D235,554

CONTINENTAL SPECIALTIES CORPORATION

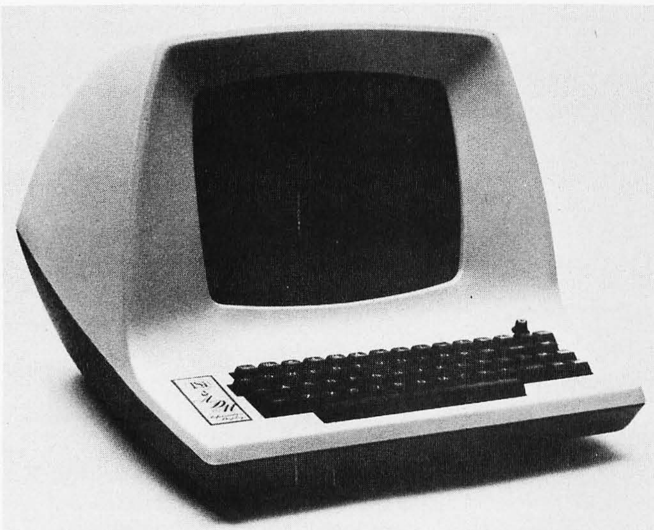
CSC

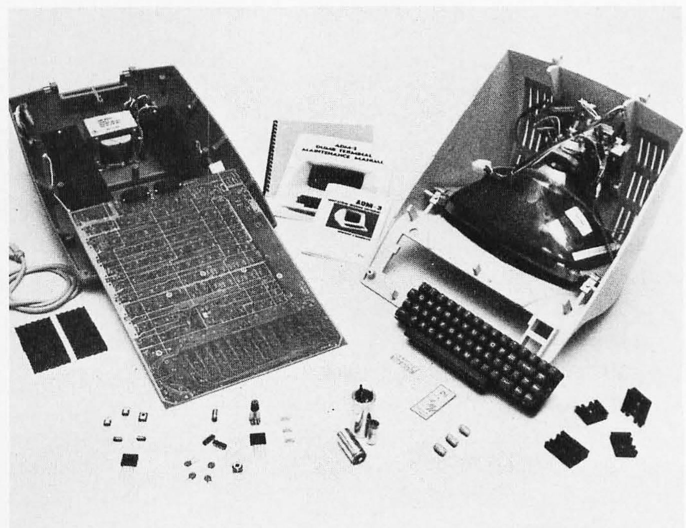
44 Kendall Street, Box 1942, New Haven, Ct 06509
203-624-3103/TWX: 710-465-1227
West Coast office: Box 7809, San Francisco, CA 94119
415-421-8872/TWX: 910-372-7992

synchro-sound enterprises

"THE COMPUTER PEOPLE"

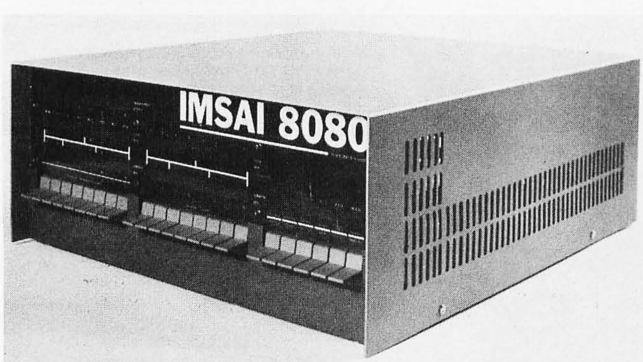
" IN STOCK "
NEW LEAR SIEGLER ADM - 3A KIT
FULL ADDRESSABLE CURSOR





CHARACTER GENERATION
5 x 7 dot matrix.
DISPLAY FORMAT
Standard: 1920 characters, displayed in 24 lines of 80 characters per line.
CHARACTER SET
Standard: 64 ASCII characters, displayed as upper case, plus punctuation and control.
COMMUNICATIONS RATES
75, 110, 150, 300, 600, 1200, 1800, 2400, 4800, 9600, 19,200 baud (switch selectable).

COMPUTER INTERFACES
EIA standard RS232C and 20 mA current-loop (switch selectable).
DATA ENTRY
New data enters on bottom line of screen; line feed causes upward scrolling of entire display with top-of-page overflow. Automatic new line switch selectable, end-of-line audible tone.



IMSAI 8080 MICROCOMPUTER
POWERFUL • EASY TO USE • LOW COST

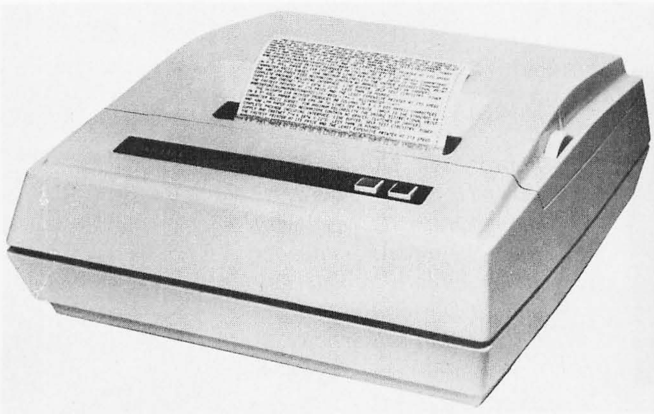
$619.95/22 SLOT MOTHER BOARD
849.95/WITH Z-80 CPU

ADM-3A KIT . . . . $ 895.00
ADM-3A ASSEMBLED 1099.95
LOWER CASE OPTION 89.00

FOLLOWING ITEMS IN STOCK TDL, OAE, PROCESSOR TECH., SWTP, APPLE, HAYDEN, TARBELL, IMSAI, LEAR SIEGLER, OKIDATA, DEC, JAVELIN, TELETYPE ASR-33, TRW.

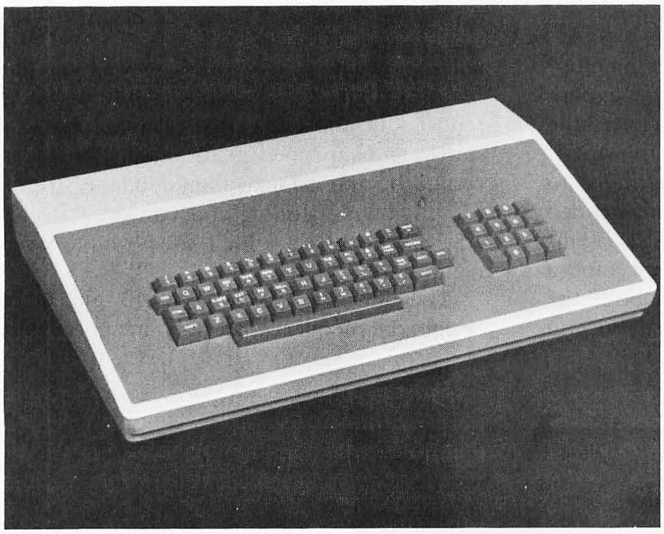
SPECIAL 9" JAVELIN HIGH RESOLUTION VIDEO MONITOR $159.95

* * * * * * * * * * * * * * * * *
* 2708 EPROM $59.95 *
* Set of seven $350.00 *
* * * * * * * * * * * * * * * * *



($1099.00)
OKIDATA MODEL 110
110 CPS DOT MATRIX LINE PRINTER

FRICTION FEED*	$1099.00
TRACTOR FEED	1229.00
RS 232C SERIAL INTERFACE	250.00



*

MOST ITEMS IN STOCK FOR SAME DAY SHIPPING. FULL MODERN REPAIR FACILITIES ON PREMISES—WE SERVICE WHAT WE SELL.

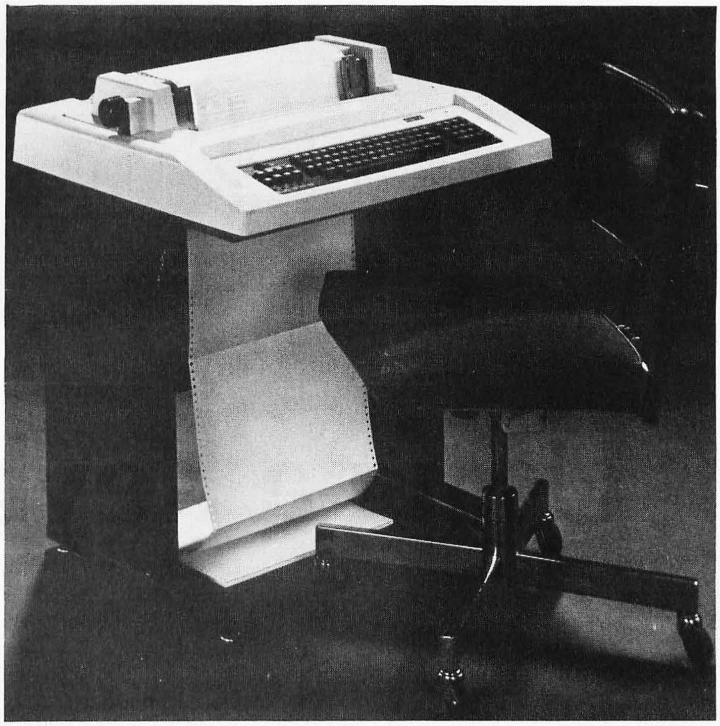
*

synchro-sound enterprises

193-25 Jamaica Ave., Jamaica, NY 11423
Phone (212) 468-7067

HOURS 9 – 4 DAILY + SATURDAY
BANKAMERICARD MASTER CHARGE
VISIT OUR NEW SHOWROOM
WORKING UNITS ON DISPLAY

DECwriter II



$1769.95

Features
132 column printing
30 CPS
Full Keyboard
Tractor Feed

NEW COMMERCIAL QUALITY KEYBOARD

The Model SS-1 Communications Terminal is a non-contact capacitive keyboard with a guaranteed life of over 100,000,000 operations.

FEATURES
- **MOS/LSI Encoder** with high reliability and low power consumption
- **n Key Roll Over** which eliminates operator error and increases thruput
- **Hysteresis** for tease proof operation
- **Solid State** performance at mechanical switch prices
- **Tactile Feedback** at the operate point
- **ASR-33 Array** with four mode encoding

KEYBOARD KIT	$ 79.95
ENCLOSURE (WITH NUMERIC CUTOUT)	39.95
NUMERIC PAD	29.95
COMPLETE KIT (with pad)	139.95
ASSEMBLED UNIT (with pad)	189.95

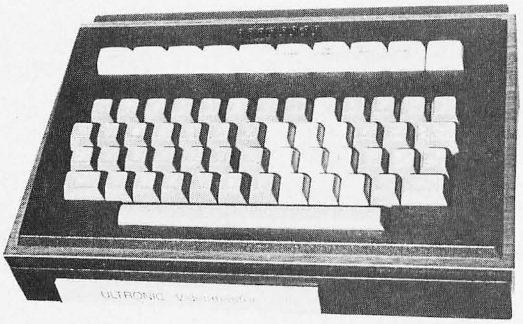


Photo 1: The author's surplus keyboard, mounted in its case with the keyboard encoder ROM circuit, repeat key and LED monitor features discussed in this article.

Using a Keyboard ROM

Bob Brehm
1095 Sherman Av
Menlo Park CA 94025

The keyboard encoder read only memory has become more plentiful in the experimenters' market recently. This versatile device can be used to generate the full ASCII character set thus allowing the hobbyist to easily and economically convert unencoded or non-ASCII keyboards to ASCII. This article presents both the logic theory and a general description of one such encoder, the SMC KR2376 MOS keyboard ROM, so you will know how it works as well as the features that are available. Addition of a repeat key and LED code readouts are also shown and explained. After reading this, you should be able to take any keyboard, strip it of unknown encoding and convert it to an ASCII keyboard for use with your computer.

The SMC KR2376 is a 40 pin MOS keyboard encoder read only memory with

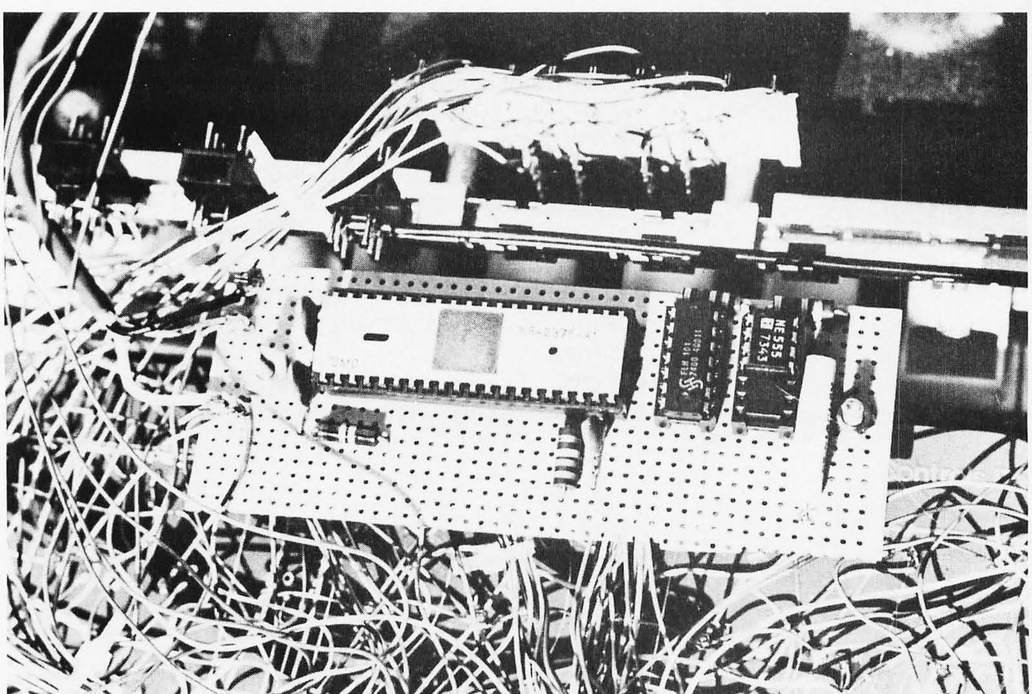


Photo 2: The keyboard encoder and related circuitry is mounted on perforated board on the back side of the keyboard assembly. The maze of wires in this picture is the result of creating the keyboard matrix by hand wiring all cross points.

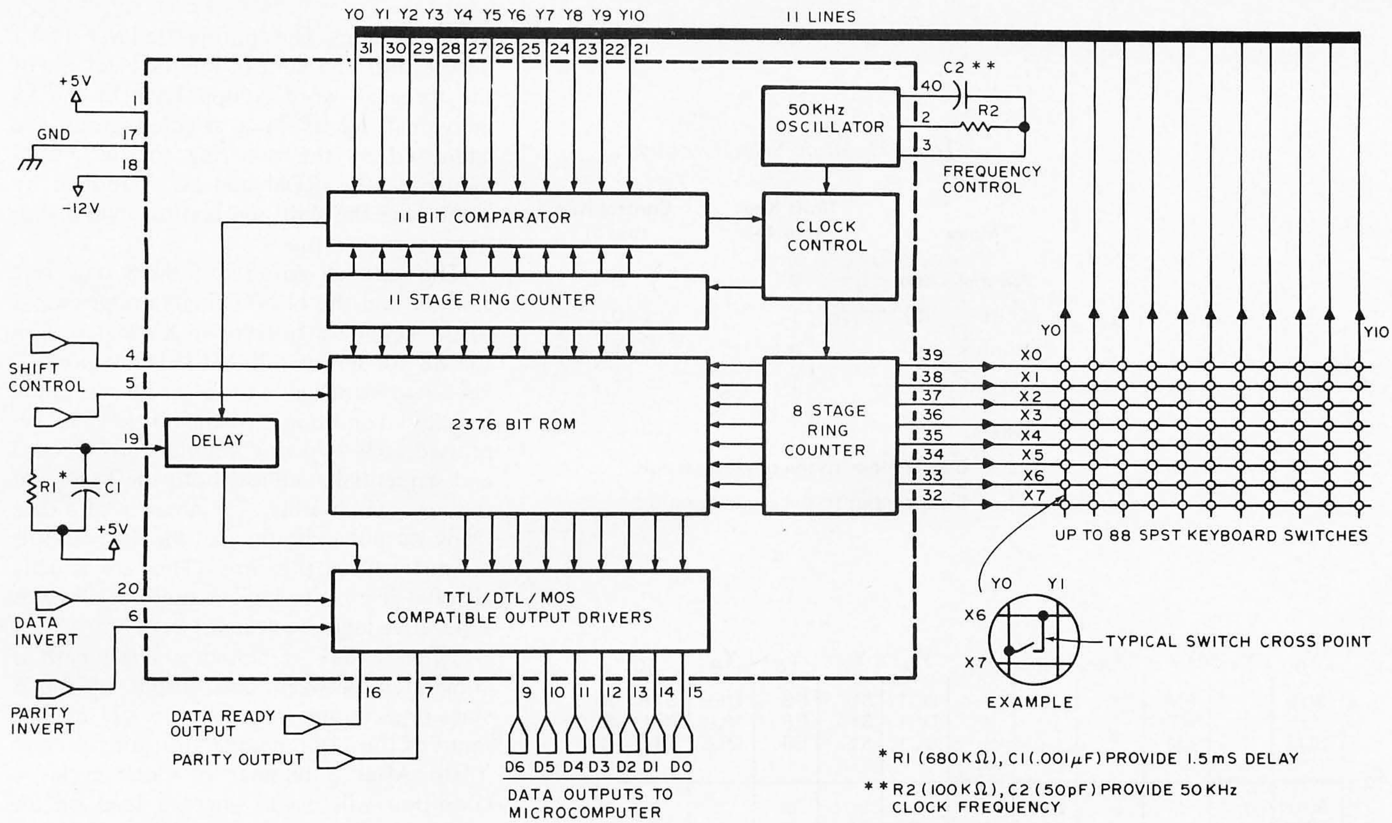


Figure 1: The internal organization of a typical keyboard encoder ROM array. The ring counters are a logic design element which generates a one of "n" output which changes each time a clock pulse is received. Thus, picking one of the 8 X output rows with one ring counter, and listening to one of the 11 Y input columns through a comparator addressed by the other ring counter, the device can select a unique cross point of the matrix. The ring counter outputs also address the ROM array, so that stopping the clocks when a key is found depressed allows the ROM to decode the ASCII output for the key contact.

all the logic necessary to encode single pole single throw keyboard switches into ASCII code. This integrated circuit features TTL/DTL or MOS compatible outputs, 2 key rollover, N key lockout, self-contained oscillator, a delay network to eliminate the effect of contact bounce, and static charge protection of all inputs and outputs! In addition external control is provided for selection of output polarity and even or odd parity generation. With prices ranging from $6 to $20, depending on source, all these features are quite a bargain.

The logic design of this ROM is easily understood if you look at the internal structure and outside world connections shown in figure 1.

The KR2376 contains a 2376 bit ROM, an 8 stage ring counter, an 11 stage ring counter, an 11 bit comparator, an oscillator circuit, an externally controllable delay network for eliminating the effect of contact bounce, and output drivers compatible with TTL, DTL or MOS logic families.

The ROM portion of the chip is a 264 by 9 bit memory arranged into three 88 word

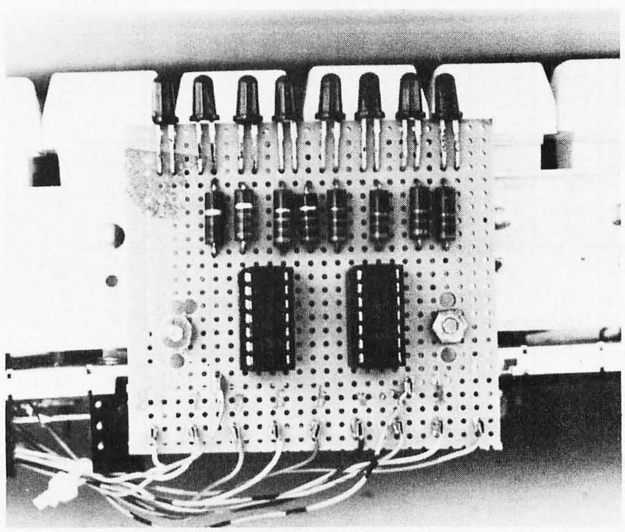


Photo 3: The LED monitor assembly is also constructed on perforated board, then mounted to the keyboard assembly. A corresponding hole in the cabinet was cut so that the LEDs are visible when the keyboard is being used (see photo 1).

Table 1: Mode Selection Logic.

Mode		Shift Key (pin 4)	Control Key (pin 5)
Normal	(N)	0	0
Shift	(S)	1	0
Control	(C)	0	1
Invalid		1	1

where:

0 = 0 volts input to the pin in question

1 = 5 volts input to the pin in question



Figure 2: Key switches hooked up to the author's keyboard encoder. This chart shows the ASCII codes available in the KR2376 ST ROM, and the wiring of switches within the matrix in the author's case. An ASCII code chart (see figure 6) can be used to verify the switch wiring by looking at bit patterns using the LED indicators described in figure 5.

by 9 bit groups. The appropriate level (+5 V) on the shift and control inputs select one of the three 88 word groups. Each of the 88 individual words in a selected group are addressed by the two ring counters. Consequently the ROM address is formed by combining the shift and control inputs with the two ring counters.

The external outputs of the 8 stage ring counter and the 11 bit comparator are wired to the keyboard to form an XY matrix with the 88 (or less if full ASCII is not needed) keyboard switches as the cross points. In the standby condition, when no key is depressed, the two ring counters are clocked and sequentially address both the ROM and the key switch array. The absence of a data ready output indicates that the data outputs are not valid at this time. (They are actually all ones if positive logic is used, or all zeros if negative logic is selected.)

When a key is closed a single path is completed between one output of the 8 stage ring counter (pins X0 thru X7) and one input of the 11 bit comparator (pins Y0 thru Y10). After a number of clock cycles, a condition will occur where a level on the selected path to the comparator matches a level on the corresponding comparator input from the 11 stage ring counter. When this event takes place, the comparator generates a signal to the clock control and to the data ready output via the externally controlled delay network. The clock control stops the clock to both the 8 and 11 bit ring counters and the data output bits (D0 to D6) stabilize

Table 2: Some Control Codes. Most keyboards do not have enough keys to generate all 88 possible cross points of the matrix shown in figure 2. As a result, certain control mode overlaps are used. The following is a table of the control functions which were implemented in the author's version of the keyboard encoder. To get these ASCII codes, press the indicated letter or key and the "CTRL" key simultaneously.

Key Combination	ASCII Code
Ctrl C	ETX
Ctrl B	STX
Ctrl +	ESC
Ctrl G	BEL
Ctrl D	EOT
Ctrl L	FF
Ctrl J	LF
Ctrl M	CR
Ctrl H	BS

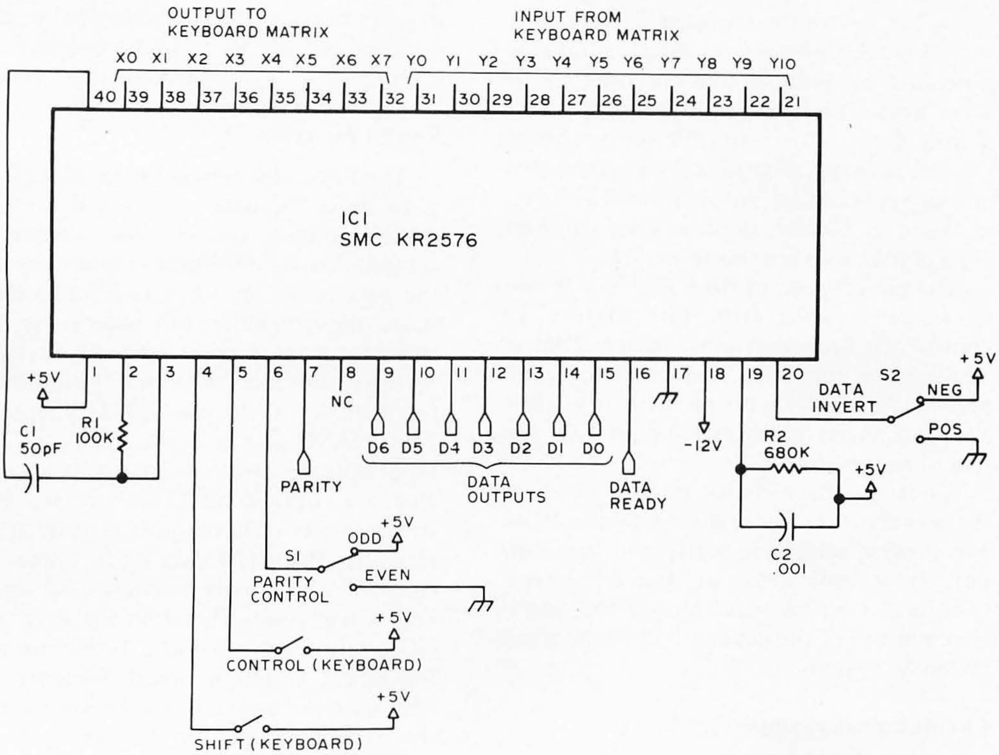


Figure 3: Specific wiring diagram for the keyboard encoder. This diagram shows the major part of a parallel keyboard interface, the encoder itself.

with the ASCII code of the depressed key. When D0 to D6 are stable a valid signal is indicated on the data ready output pin. The data outputs remain stable until the key is released.

An additional feature provides for external polarity control of the data outputs. Parity invert provides polarity control of the parity output while the data (and data ready) invert input provides for polarity control of the data outputs D0-D6 and the data ready output. Both of these features are very useful depending upon your computer requirements for odd or even parity and positive or negative data outputs.

The above general description should provide you with enough knowledge to understand a practical implementation of the read only memory in a typical surplus keyboard. The trick is to remove the original encoder and to wire up the new, documented encoder, using figure 1 as a model.

Implementation

The keyboard described is an "Ultronic Videomatic 7000" which was purchased from John Meshna for about $35. While I used this specific surplus board, any keyboard which uses SPST switches can be adapted to a new encoder. Do not attempt to use "Hall effect," capacitive sensing or other "no switch" keyboards with this method. This keyboard has a nice metallic enclosure which helps shield the read only memory and its associated circuitry from noise. The diode board that came with the unit was discarded to make room for the ROM board and the LED monitoring readouts.

My keyboard had all switches soldered to a circuit board with one side of each switch common to ground. It was necessary to break the foil pattern and hook the switches up in a manner consistent with the ROM requirements. Many newer keyboards already have their switches wired properly for this ROM and need not be changed. The XY matrix wiring chart for the full ASCII character set is shown in figure 2. To wire your own keyboard, connect one side of all switches of each Y_i column together; eg: for Y_4 I connected together one side of the switches for M, =, H, Y, #. The other side of the switches go to the X_i row buses which go to the appropriate pins on the ROM.

Each X_i and Y_i switch can generate three different codes depending upon the voltages present on the STROBE (S) and CONTROL (C) inputs. Mode selection logic is shown in table 1.

As can be seen from figure 2 there exist several redundant keys whose codes may be generated by enabling the control code of other keys. This characteristic is very useful if you don't have a full 88 key keyboard (the usual case). Actual keys implemented on my keyboard are shown as small switches in figure 2. Several of these keys and their control functions are shown in table 2.

The pin diagram of the KR2376 is shown in figure 3 along with the circuitry to control the "function pins" of the ROM in addition to the delay and oscillator components. On my keyboard the parity select and data invert switches are placed on the rear of the enclosure.

So far we have shown how to hook up the switches on the keyboard to the ROM and provide selectable parity and data output. It is now time to describe several additional features that enhance the operation and use of the keyboard in the personal computer system.

Circuit Enhancements

You can enhance the usefulness of your keyboard with the addition of a repeat key, selectable strobe output circuitry and an LED code readout circuit. Each of these improvements makes your keyboard more versatile and allows it to be connected to computers with varying input requirements.

Repeat Function

The basic idea behind the repeat function is to pulse the data ready signal while the ASCII outputs remain stable. Since the computer reads a keyboard input every time the data ready line changes from its steady state, the computer will accept the same character many times if we pulse this signal. An easy and economical way to implement this function is to use a 555 timer and a 7400 NAND gate as shown in figure 4. The timer operates at approximately 10 Hz or 10 characters per second. Closure of the repeat switch allows IC3b output to change at a 10 Hz rate. The ROM data ready output is a constant +5 V (using positive logic) when a key is depressed. Therefore the data ready (IC3c output) is gated at a 10 Hz rate when the repeat switch is closed. Since we have only used three gates of the 7400 we can use the remaining gate to implement a data ready invert function useful if a positive going strobe is required with positive logic. Additional combinations can be made with negative logic. To use the repeat function

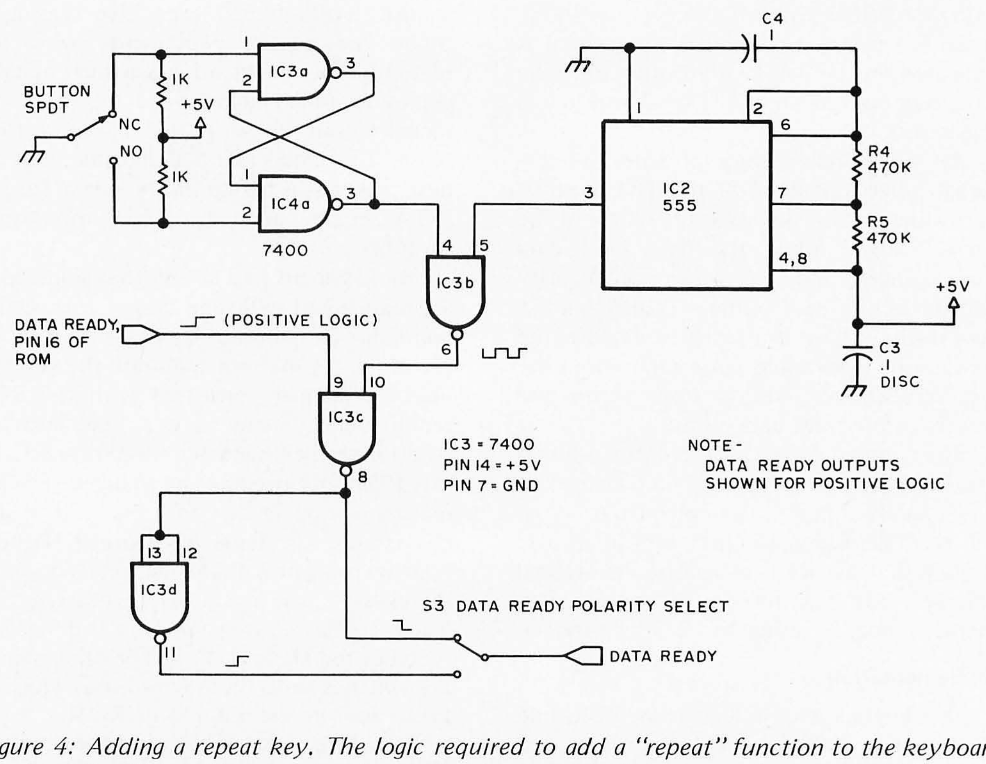


Figure 4: Adding a repeat key. The logic required to add a "repeat" function to the keyboard consists of a 555 oscillator and gating under the control of the repeat key switch. This circuit also provides for either normal or inverted forms of the data ready line output. A key is "repeated" by making the normal steady state output of the keyboard encoder's data ready line oscillate at about a 10 Hz rate.

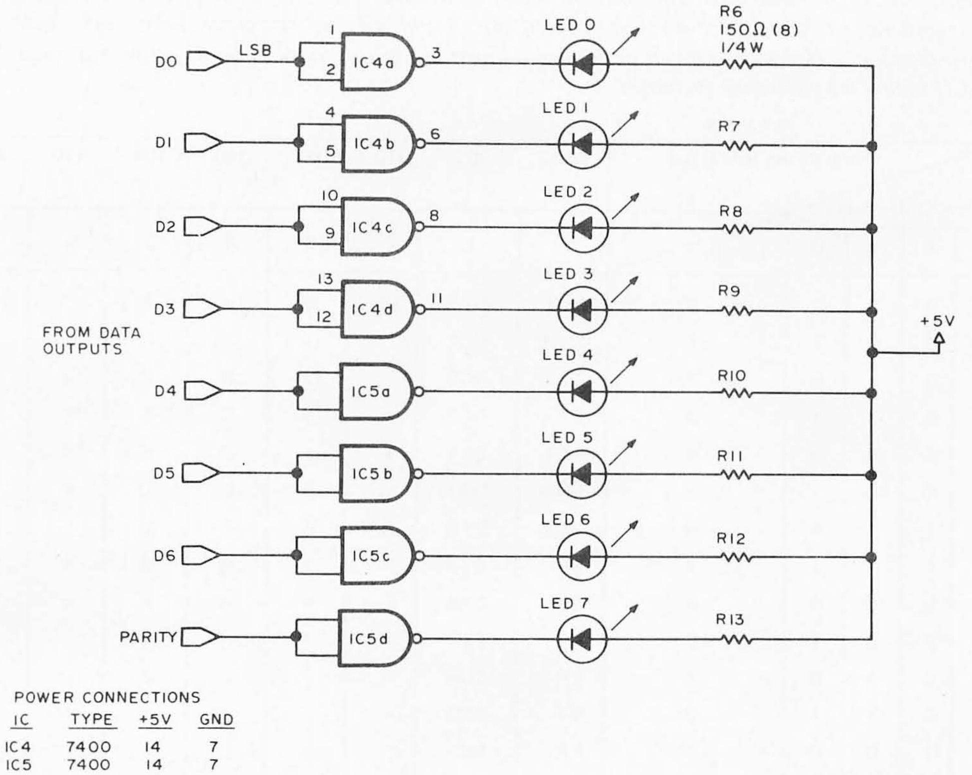


Figure 5: Keyboard monitor LED display circuit. Adding two 7400 NAND packages, 8 resistors and 8 LED indicators provides an invaluable debugging tool for checking out the keyboard matrix wiring — and a fascinating display once the wiring is pronounced error free. Check out each key in normal, shifted and control modes using the standard ASCII code chart of figure 6.

just press the desired alphanumeric key and then the repeat key until the desired number of letters are transmitted.

LED Code Display

This option is by no means necessary but is very useful to check out the operation of the keyboard as well as have a ready source for determining the ASCII code of any keyboard character. The circuit diagram is shown in figure 5.

When any input to the 7400 NAND gate goes high it forces the output low which grounds the LED causing it to light. During standby using positive logic all inputs will be high and consequently all LEDs will be lit. Negative logic will cause them to be extinguished in standby. The LEDs are arranged on the keyboard such that the rightmost LED represents the least significant bit (D0) and the leftmost LED represents the parity bit. Using positive logic the ASCII code for any depressed key will be shown on the LEDs as a lit LED for a 1 and a dark LED for a 0. The opposite is true for negative logic. It is very interesting to watch the parity bit change as the code changes. With even parity you should always have an even number of LEDs lit and with odd parity an odd number should be lit.

Final Checkout

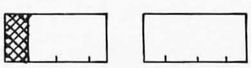
Final checkout should determine whether the proper ASCII code is present on the LEDs (thus the output), whether the shift, control, parity select, data ready select, and data invert options function properly, and whether the repeat function actually pulses the strobe signal.

Code checkout is easy if you refer to the ASCII code chart given in figure 6. All keys that you have implemented should show their codes on the LEDs. (In the event you didn't install the LED readouts, a voltmeter at every bit will show the logic levels. Individual bit measuring for 264 codes is very tedious and since the cost of the LED readout unit is only about $3 it is highly recommended that this feature be built for test purposes if not actually installed.) Try all keys in the normal (lower case ASCII letters and numbers), shift (upper case ASCII and shift characters) and the control mode. Press each key several times to make sure the code is repeated on the LEDs. In my case I found that for

Figure 6: USA Standard for Information Exchange (ASCII) code chart. The physical arrangement of key tops on your particular keyboard switch array may vary, but the correspondence between a given code and a given graphic symbol should follow this chart for ASCII character exchange purposes.

Low Order Bits				High Order Bits 6,5,4	000	001	010	011	100	101	110	111
3	2	1	0	Hex 0 \ Hex 1	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	"!"	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	A	LF	SUB	*	:	J	Z	j	z
1	0	1	1	B	VT	ESC	+	;	K	[	k	{
1	1	0	0	C	FF	FS	,	<	L	\	l	¦
1	1	0	1	D	CR	GS	—	=	M	]	m	}
1	1	1	0	E	SO	RS	.	>	N	↑	n	~
1	1	1	1	F	SI	US	/	?	O	←	o	DEL

Typical
8 bit
Hexadecimal
Format



7 6 5 4   3 2 1 0

HIGH ORDER 1   LOW ORDER 0

Bit 7 is ignored in 8 bit processors.

The following table gives expansions of the multiple letter mnemonic functions in the chart.

- NUL Null
- SOH Start Of Heading
- STX Start of TeXt
- ETX End of TeXt
- EOT End Of Transmission
- ENQ ENQuiry
- ACK ACKnowledge
- BEL BELl
- BS Back Space
- HT Horizontal Tab
- LF Line Feed
- VT Vertical Tab
- FF Form Feed
- CR Carriage Return
- SO Shift Out
- SI Shift In
- DLE Data Link Escape
- DC1, DC2, DC3, DC4 Peripheral control
- NAK Negative AcKnowledge
- SP Space
- SYN SYNchronous idle
- ETB End Transmission Block
- CAN CANcel
- EM End of Medium
- SUB SUBstitute
- ESC ESCape
- FS File Separator
- GS Group Separator
- RS Record Separator
- US Unit Separator
- DEL DELete (rubout)

certain letters I would get multiple codes. This problem was traced to having several X_i lines hooked together or Y_i lines hooked together. After correcting these errors the keyboard functioned properly. Voltages should not exceed +5 V for input on pin 1 or −12 V on pin 18 of the ROM. Total power consumption is 200 mW for the ROM, 265 mW for the TTL ICs, and 1333 mW for the LEDs.

Interfacing the keyboard to your computer should be easy if a parallel input port is available. Simply connect the D0 to D6 inputs, the parity line and the data ready line to the input port using the appropriate logic level as required by your computer.

If you have a serial input port available for your keyboard, a parallel to serial converter is necessary. You can use half of a UART as discussed in Don Lancaster's excellent article "Serial Interface" in the September 1975 issue of BYTE [*page 22*]. This strategy, however, leaves half of the UART unused. A more reasonable approach for serial interface is to use the Teletype transmitter circuit which is described in full detail in Don Lancaster's *TTL Cookbook* published by Howard W Sams, Indianapolis IN, and available from BITS Inc.

Building an ASCII encoded keyboard using the KR2376 ROM is easy and economical, and, in addition, provides a full character set usually found only in expensive keyboard logic assemblies.■


FD-8 DISK MEMORY
Midwest Scientific Instruments Olathe, Kansas
6800 COMPUTER SYSTEM
Diskette

Implementing the Tiny Assembler

Jack Emmerichs
8465 N 51st
Brown Deer WI 53223

When all the functional requirements for Tiny Assembler had been described in a well structured format, the final structural network of the program was realized, as given at the end of last month's introduction to this program. Combining figure 3 of page 66, April 1977 BYTE, with detailed table definitions, I was at long last ready to begin actual coding of the assembler itself.

Coding the Program

I was immediately confronted with two problems. First, I did not yet have an assembler to work with. Second, I had no M6800 machine available! I therefore decided to start by writing a cross assembler in PL/I. This would let me find and correct any logical bugs in the assembler's design. It would also provide an assembler capable of producing the final M6800 version of the developed program.

Because structured design techniques are language independent, no redesign was necessary for the development of the cross assembler. (In fact, assemblers written in 8080 or Z-80 code could be developed just as easily.) During testing of this program, several minor logical errors were corrected, and an operational design was available for the final version of the program. In a very real sense, the assembler was used to write and debug itself.

The next step was to translate each function from the PL/I code to M6800 assembler code. As each block of code was developed, it was added to existing functions and run through the cross assembler to catch any errors and to find the length of the generated code. Final debugging of each function had to be postponed until a machine was available. The code was basically developed in a top down manner starting with the mainline and working down the hierarchy of functions. However, the order of listing different parts of the program was

Listing 1: The external address table. This series of jumps is the sole interface to the outside world from Tiny Assembler 6800. As originally written, terminal oriented IO is done with these five external references to MIKBUG, Motorola's Monitor Program.

```
                         | *       ****************** EXTERNAL ADDRESS TABLE
                         | *
(0790)   7E   E1   D1    | CHROUT JMP   $E1D1
(0793)   7E   E1   AC    | CHRIN  JMP   $E1AC
(0796)   36              | HEXOUT PSHA
(0797)   BD   E0   67    |        JSR   $E067
(079A)   32              |        PULA
(079B)   7E   E0   6B    |        JMP   $E06B
(079E)   7E   E0   CC    | BLKOUT JMP   $E0CC
(07A1)   7E   E0   7E    | STROUT JMP   $E07E
```

Notes on External Linkages

If MIKBUG is used (for example, in a Southwest Technical Products machine, or an expanded version of one of Motorola's development board kits) then this code should be used as is. If you interface to your own custom monitor, this code must be replaced by references to equivalent routines as follows:

CHROUT: The contents of the accumulator, treated as an ASCII character, are dumped on the output device.
CHRIN: The contents of the accumulator are defined by an ASCII code obtained from the input device.
HEXOUT: The contents of accumulator A are dumped to the output device as two hexadecimal encoded ASCII digits.
BLKOUT: A blank is dumped on the output device.
STROUT: A string, terminated by an EOT (hexadecimal 04) code is dumped on the output device. The index register points to the string on entry, and is updated during the operation.

M6800 Structured Code Conventions

	General Structure	M6800 Implementation
IF THEN ELSE	IF (test condition) THEN : (then processing) ELSE : (else processing) ENDIF	IF TST CONDITION BNE ELSE *** } then processing *** BRA ENDIF ELSE *** } else processing *** ENDIF *** } continue ***
DO WHILE	(set initial cond) DO WHILE (condition) : (do group) : (may change cond) ENDDO	* SET INITIAL COND DO TST CONDITION BNE ENDDO *** } do group processing *** *** BRA DO ENDDO *** ***
DO UNTIL	(set initial cond) DO UNTIL (condition) : (do group) : (may change cond) ENDDO	* SET INITIAL COND DO *** } do group processing *** *** TST CONDITION BNE DO ENDDO *** ***
CASE	A general case structure: SELECT (based on case) : CASE (condition 1) : : (process 1) : CASE (condition 2) : : (process 2) : CASE (condition 3) : : (process 3) ENDSELECT The case structure, recoded using the DO UNTIL to check case: SELECT : (point to first case) : DO UNTIL (selected condition) : : (advance to next case) : ENDDO : (execute case pointed to) ENDSELECT	JSR SELECT JMP ENDSELECT SELECT LDX #CASETABL LOOP JSR NXTCASE Calculate condition TST CONDITION BEQ EXECUTE INX INX BRA LOOP Move to next CASE EXECUTE JSR X Execute CASE RTS Return from select * CASETBL BRA CASE1 } Table of CASE Conditions BRA CASE2 BRA CASE3 * CASE1 *** } First CASE procedure *** RTS CASE2 *** } Second CASE procedure *** RTS CASE3 *** } Third CASE procedure *** RTS ENDSELECT *** } continue *** ***

*Table 1: The general programming structures shown in the table above have been implemented as shown in the right hand column where *** represents the particular code which is executed within these control structures.*

influenced by the one pass design of the assembler. All data declarations, output text, tables, and commonly called subroutines were placed at the front of the listing to reduce the number of forward references. The program mainline follows the section containing common subroutines.

While the M6800 has a very powerful instruction set, it cannot directly support the functions shown in the pseudocode used in the listings given as examples. Therefore, control functions such as DO, THEN, ELSE, UNTIL, WHILE and CASE need to be performed by in line routines of M6800 instructions. Table 1 shows some of the common functional structures with their associated assembly language routines. The basic structures of these in line routines will

Text continued on page 92

Table 2: Hexadecimal object code. This table contains the complete object code of Tiny Assembler 6800, assembled at an origin of 0000 hexadecimal in memory address space. This version can be used if necessary to load Tiny Assembler 6800 by hand. This type was set by machine from the same data which was used to create the bar code listings of figure 1. Note, however, that to make this version readable, fixed length lines were used (as opposed to variable length lines in figure 1).

Address	Data															
0000	7E	07	A4	00	20	00	00	00	00	20	20	20	00	00	00	00
0010	00	00	2A	20	20	20	00	0D	00	04	00	0D	00	00	00	00
0020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0040	58	20	20	20	0D	00	00	2A	2A	2A	20	45	52	52	4F	52
0050	20	2D	20	00	0A	00	00	20	04	0D	0A	0A	0A	0A	00	00
0060	4C	4F	43	4E	20	42	31	20	42	32	20	42	33	20	0A	0A
0070	00	00	04	0D	00	00	12	04	0D	0A	0A	0A	00	00	2A	2A
0080	2A	20	55	4E	52	45	53	4F	4C	56	45	44	20	49	54	45
0090	4D	53	3A	0D	0A	00	00	04	0D	0A	00	00	2A	2A	2A	20
00A0	53	59	4D	42	4F	4C	53	2C	20	53	4F	52	54	3F	0D	0A
00B0	0A	00	04	00	00	20	20	20	00	00	00	00	00	00	00	00
00C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00F0	00	00	06	06	11	17	08	1F	04	23	03	26	03	00	00	00
0100	00	29	03	2C	02	00	00	2E	04	00	00	32	02	34	02	36
0110	02	00	00	38	05	3D	0A	47	07	00	00	00	00	4E	01	00
0120	00	00	00	00	00	42	41	1B	50	44	43	89	10	44	44	8B
0130	10	4E	44	84	10	53	4C	48	30	53	52	47	30	43	43	24
0140	40	43	53	25	40	45	51	27	40	47	45	2C	40	47	54	2E
0150	40	48	49	22	40	49	54	85	10	4C	45	2F	40	4C	53	23
0160	40	4C	54	2D	40	4D	49	2B	40	4E	45	26	40	50	4C	2A
0170	40	52	41	20	40	53	52	8D	40	56	43	28	40	56	53	29
0180	40	42	41	11	50	4C	43	0C	50	4C	49	0E	50	4C	52	4F
0190	30	4C	56	0A	50	4D	50	81	10	4F	4D	43	30	50	58	8C
01A0	11	41	41	19	50	45	43	4A	30	45	53	34	50	45	58	09
01B0	50	4E	44	00	E0	4F	52	88	10	51	55	00	F0	43	42	01
01C0	F0	43	43	02	F0	44	42	03	F0	4E	43	4C	30	4E	53	31
01D0	50	4E	58	08	50	4D	50	4E	31	53	52	8D	31	44	41	86
01E0	10	44	53	8E	11	44	58	CE	11	53	52	44	30	45	47	40
01F0	30	4F	50	01	50	52	41	8A	10	52	47	04	F0	53	48	36
0200	00	55	4C	32	00	4D	42	05	F0	4F	4C	49	30	4F	52	46
0210	30	54	49	3B	50	54	53	39	50	42	41	10	50	42	43	82
0220	10	45	43	0D	50	45	49	0F	50	45	56	0B	50	54	41	87
0230	20	54	53	8F	21	54	58	CF	21	55	42	80	10	57	49	3F
0240	50	41	42	16	50	41	50	06	50	42	41	17	50	50	41	07
0250	50	53	54	4D	30	53	58	30	50	58	53	35	50	41	49	3E
0260	50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0270	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0290	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
02A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
02B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
02C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
02D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01	01
02E0	01	01	39	00	00	00	00	00	00	00	00	00	00	00	00	00
0780	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0790	7E	E1	D1	7E	E1	AC	36	BD	E0	67	32	7E	E0	6B	7E	E0
07A0	CC	7E	E0	7E	BD	02	DE	CE	02	61	6F	00	08	8C	07	8F
07B0	26	F8	86	39	B7	02	DE	CE	00	B8	DF	2A	7F	00	1C	7F
07C0	00	31	CE	07	B7	FF	A0	48	8E	A0	7F	CE	00	59	BD	07
07D0	A1	BD	09	2A	BD	0F	5A	7E	E0	E3	CE	00	B8	86	3E	BD
07E0	07	90	BD	07	93	81	21	26	08	8C	00	B8	27	F4	09	20
07F0	F1	A7	00	91	1B	26	01	39	8C	00	F0	27	E5	81	20	2B
0800	E1	81	5F	22	DD	08	20	DA	DF	24	DE	2A	96	17	91	1B
0810	26	14	7D	00	31	27	03	BD	08	B2	86	01	97	31	BD	07
0820	DA	CE	00	B8	DF	2A	A6	00	97	17	08	DF	2A	DE	24	39
0830	96	17	91	0A	27	01	39	BD	08	08	20	F6	DE	0A	DF	05
0840	DF	07	CE	00	05	D6	17	D1	0A	27	1D	D1	1B	27	19	C1
0850	2B	27	15	C1	2D	27	11	C1	2C	27	0D	E7	00	8C	00	08
0860	27	01	08	BD	08	08	20	DD	39	97	53	DF	20	CE	00	44
0870	BD	07	A1	DE	20	7C	00	03	39	7D	00	03	23	01	39	96
0880	16	D6	31	C1	01	27	19	C1	02	26	07	97	B6	7C	00	31
0890	20	1A	C1	03	26	07	97	B7	7C	00	31	20	0F	BD	08	B2
08A0	DE	2F	DF	B3	96	16	97	B5	86	02	97	31	DE	2F	08	DF
08B0	2F	39	DF	24	CE	00	73	BD	07	A1	CE	75	30	6D	00	09
08C0	26	FB	86	01	BD	07	90	96	B3	BD	07	96	96	B4	BD	07
08D0	96	C6	01	CE	00	B5	D1	31	2B	17	26	0A	86	01	BD	07
08E0	90	86	14	BD	07	90	BD	07	9E	BD	07	9E	BD	07	9E	20
08F0	08	BD	07	9E	A6	00	BD	07	96	08	5C	C1	04	26	D7	96
0900	31	81	04	26	0A	86	01	BD	07	90	86	14	BD	07	90	BD
0910	07	9E	DE	2F	DF	B3	DE	24	86	0A	BD	07	90	96	0A	97
0920	B5	97	B6	97	B7	86	01	97	31	39	CE	00	04	D6	0A	7F
0930	00	03	7F	00	3F	E7	00	08	8C	00	0A	26	F8	96	1B	97
0940	17	BD	08	08	91	1B	27	E2	81	2A	27	DE	91	0A	27	09
0950	BD	08	3C	7C	00	26	BD	0D	1B	BD	08	30	91	1B	27	3D
0960	BD	08	3C	86	02	97	19	BD	09	DD	86	04	97	19	7D	00
0970	03	22	B7	96	18	81	0E	26	04	BD	08	B2	39	D6	2E	81
0980	0F	26	05	BD	0E	DB	20	15	BD	08	30	81	23	26	05	97
0990	04	BD	08	08	BD	08	30	BD	09	A3	BD	0A	CA	BD	0E	15
09A0	7E	09	2A	DE	2A	96	17	81	58	26	1E	A6	00	91	0A	27
09B0	0E	91	1B	27	0A	81	2B	27	06	81	2D	27	02	20	0A	86
09C0	58	97	04	86	30	97	17	20	13	A6	00	08	91	0A	27	0C
09D0	91	1B	27	08	81	2C	26	F1	A6	00	97	04	39	96	05	81
09E0	41	2C	06	86	41	BD	08	69	39	81	5A	2F	06	86	41	BD
09F0	08	69	39	84	1F	4A	48	CE	00	F1	4D	27	04	08	4A	20
0A00	F9	E6	01	D7	29	E6	00	4F	58	49	58	49	CE	01	25	DF
0A10	3D	DB	3E	99	3D	D7	3E	97	3D	CE	00	06	DF	32	7D	00
0A20	29	26	06	86	42	BD	08	69	39	DE	3D	DF	34	BD	0A	A5
0A30	7D	00	11	23	09	BD	0A	4E	7D	00	11	23	01	39	08	08
0A40	08	08	DF	3D	CE	00	06	DF	32	7A	00	29	20	D0	A6	02
0A50	97	2E	A6	03	16	C4	0F	D7	1D	44	44	44	44	97	18	D6
0A60	08	D1	0A	26	0B	7D	00	1D	22	3A	81	02	23	33	20	34
0A70	81	03	22	2D	7D	00	1D	22	28	C1	41	27	19	C1	42	26
0A80	20	D6	2E	4D	26	06	CB	01	D7	2E	20	0A	81	03	27	02
0A90	CB	30	CB	10	D7	2E	4C	85	02	26	09	86	05	97	18	20
0AA0	03	7F	00	11	39	36	DF	20	9F	38	7F	00	11	96	19	97
0AB0	3C	DE	32	9E	34	34	32	A1	00	26	09	08	7A	00	3C	26
0AC0	F5	7C	00	11	DE	20	9E	38	32	39	96	18	D6	2E	81	03
0AD0	23	0F	D7	16	BD	08	79	96	18	81	04	26	03	BD	0B	05
0AE0	39	96	04	81	23	26	04	BD	0B	4F	39	81	58	26	06	CB
0AF0	20	BD	0B	8B	39	91	0A	26	06	CB	10	BD	0B	9D	39	86
0B00	44	BD	08	69	39	BD	0B	D5	7D	00	3F	23	08	96	0D	97
0B10	16	BD	08	79	39	CE	00	2F	96	0C	D6	0D	E0	01	A2	00
0B20	C0	01	92	2C	D7	16	BD	0B	2F	01	01	BD	08	79	39	91
0B30	2C	27	06	81	FF	27	02	20	04	98	16	2A	11	86	47	BD
0B40	08	69	DE	2F	7D	00	1E	22	01	09	DF	2F	DF	B3	39	96
0B50	18	81	01	27	06	86	43	BD	08	69	39	37	BD	0B	D5	33
0B60	D7	16	BD	08	79	96	05	81	27	27	04	BD	0B	77	39	96
0B70	06	97	16	BD	08	79	39	7D	00	1D	23	07	96	0C	97	16
0B80	BD	08	79	96	0D	97	16	BD	08	79	39	37	BD	0B	D5	33
0B90	D7	16	BD	08	79	96	0D	97	16	BD	08	79	39	37	BD	0B
0BA0	D5	33	7F	00	3C	7D	00	0C	26	0D	7D	00	3F	22	08	96
0BB0	18	81	03	27	02	20	05	CB	20	7C	00	3C	D7	16	BD	08
0BC0	79	7D	00	3C	27	07	D6	0C	D7	16	BD	08	79	D6	0D	D7
0BD0	16	BD	08	79	39	7F	00	26	BD	0C	2D	BD	0C	1A	96	17
0BE0	81	2B	27	05	81	2D	27	01	39	97	09	DE	0C	DF	0E	BD
0BF0	08	08	BD	0C	2D	BD	0C	1A	96	09	81	2B	27	0E	96	0F
0C00	90	0D	97	0D	96	0E	92	0C	97	0C	20	D2	96	0F	9B	0D
0C10	97	0D	96	0E	99	0C	97	0C	20	C4	7D	00	03	22	0D	7D
0C20	00	1F	22	08	7D	00	10	23	03	BD	0D	1B	39	7C	00	10
0C30	7C	00	1F	7F	00	0C	7F	00	0D	96	17	81	24	26	0A	BD
0C40	08	08	BD	08	3C	BD	0C	DD	39	7F	00	1F	81	26	26	03
0C50	BD	08	08	BD	08	3C	CE	00	05	A6	00	91	0A	27	08	81
0C60	30	2B	3D	81	39	22	39	91	0A	27	31	8C	00	09	27	2C
0C70	85	30	26	06	86	4A	BD	08	69	39	84	0F	A7	00	C6	09
0C80	D7	3C	D6	0D	96	0C	DB	0D	99	0C	7A	00	3C	26	F7	EB
0C90	00	99	2C	D7	0D	97	0C	08	A6	00	20	CB	7F	00	10	39
0CA0	CE	00	40	DF	32	CE	00	05	DF	34	BD	0A	A5	7D	00	11
0CB0	23	0A	DE	2C	DF	05	DF	07	7F	00	10	39	CE	00	12	DF
0CC0	32	CE	00	05	DF	34	BD	0A	A5	7D	00	11	23	0E	DE	2F
0CD0	96	18	81	04	26	01	09	DF	0C	7F	00	10	39	CE	00	05
0CE0	A6	00	91	0A	27	34	8C	00	09	27	2F	81	30	2B	06	81
0CF0	46	22	02	20	06	86	4A	BD	08	69	39	81	41	2B	02	8B
0D00	09	84	0F	A7	00	96	0C	D6	0D	58	49	58	49	58	49	58
0D10	49	EB	00	D7	0D	97	0C	08	20	C6	39	96	05	81	27	26
0D20	01	39	81	40	22	06	86	4B	BD	08	69	39	CE	02	DF	C6
0D30	01	C1	C9	26	06	86	45	BD	08	69	39	DF	34	DF	3A	CE
0D40	00	05	DF	32	BD	0A	A5	7D	00	11	23	2C	DE	3A	7D	00
0D50	26	22	0C	EE	04	DF	0C	9C	2C	26	03	BD	0D	B3	39	EE
0D60	04	9C	2C	27	06	86	46	BD	08	69	39	BD	0D	95	7C	00
0D70	1E	DE	3A	EE	04	DF	0C	39	DE	3A	6D	00	26	0C	BD	0D
0D80	95	7D	00	26	22	03	BD	0D	B3	39	DE	3A	08	08	08	08
0D90	08	08	5C	20	9C	DE	3A	9F	38	9E	05	AF	00	9E	07	AF
0DA0	02	7D	00	26	23	06	9E	2F	AF	04	DF	27	EE	04	DF	0C
0DB0	9E	38	39	86	01	7F	00	3C	CE	02	61	81	19	2F	06	86
0DC0	49	BD	08	69	39	81	19	26	08	86	48	BD	08	69	7F	00
0DD0	03	E6	00	26	06	E6	01	26	02	20	0E	4C	08	08	08	08
0DE0	08	D6	3C	CB	05	D7	3C	20	D2	96	04	81	58	26	02	6A
0DF0	04	81	23	26	07	7D	00	1D	22	02	6A	04	96	18	81	04
0E00	26	02	6C	04	9F	38	9E	3A	AF	00	9E	2F	31	AF	02	9E
0E10	38	7C	00	3F	39	7D	00	1E	22	01	39	86	01	CE	02	61
0E20	DF	22	7D	00	03	22	07	81	19	2F	04	7F	00	1E	39	36
0E30	7D	00	1C	23	05	BD	0E	A9	20	61	DE	22	A6	00	E6	01
0E40	91	27	26	57	D1	28	26	53	DE	2F	DF	36	BD	08	B2	DE
0E50	22	EE	02	DF	2F	DE	22	6D	04	2B	0B	26	14	DE	27	E6
0E60	04	D7	16	BD	08	79	DE	27	E6	05	D7	16	BD	08	79	20
0E70	1A	DE	27	A6	04	E6	05	CE	00	2F	E0	01	A2	00	DE	2F
0E80	09	DF	2F	D7	16	BD	0B	2F	BD	08	79	DE	36	DF	2F	DE
0E90	22	6F	00	6F	01	6F	02	6F	03	6F	04	DE	22	08	08	08
0EA0	08	08	DF	22	32	4C	7E	0E	22	DE	22	E6	00	26	05	E6
0EB0	01	26	01	39	EE	00	86	04	97	3C	A6	00	BD	07	90	08
0EC0	7A	00	3C	26	F5	BD	07	9E	DE	22	A6	02	BD	07	96	A6
0ED0	03	BD	07	96	CE	00	93	BD	07	A1	39	CE	0F	4E	5D	27
0EE0	05	08	08	5A	20	F8	BD	08	30	AD	00	39	BD	0B	D5	9F
0EF0	38	9E	0C	DE	27	AF	04	9F	B3	9E	38	39	7F	00	1D	20
0F00	06	7F	00	1D	7C	00	1D	BD	0B	D5	BD	0B	77	96	17	81
0F10	2C	27	01	39	BD	08	08	20	EE	BD	0B	D5	96	0D	97	3C
0F20	26	01	39	BD	08	08	97	16	BD	08	79	7A	00	3C	20	F0
0F30	BD	0B	D5	DE	0C	DF	2F	DF	B3	39	DE	2F	DF	B3	BD	0B
0F40	D5	96	30	D6	2F	9B	0D	D9	0C	97	30	D7	2F	39	20	9C
0F50	20	AA	20	C5	20	AB	20	D8	20	E0	CE	00	78	BD	07	A1
0F60	CE	00	6E	BD	07	A1	7C	00	1C	86	01	97	1E	BD	0E	15
0F70	BD	0F	79	86	0A	BD	07	90	39	CE	00	98	BD	07	A1	BD
0F80	07	93	81	59	26	03	BD	0F	C8	CE	02	DF	86	05	97	3C
0F90	DF	20	CE	00	93	BD	07	A1	DE	20	86	04	97	1A	A6	00
0FA0	27	25	BD	07	90	08	7A	00	1A	26	F3	BD	07	9E	A6	00
0FB0	BD	07	96	08	A6	00	BD	07	96	08	BD	07	9E	BD	07	9E
0FC0	7A	00	3C	26	D5	20	C5	39	CE	02	DF	7F	00	11	20	12
0FD0	C6	06	A6	00	A1	06	2D	06	26	12	08	5A	20	F4	08	5A
0FE0	26	FC	6D	06	26	EA	7D	00	11	26	DD	39	D7	1A	D7	11
0FF0	A6	00	E6	06	A7	06	E7	00	08	7A	00	1A	26	F2	20	E2

Notes on Memory to Memory Operation
(Personal use notes by Carl Helmers)

As used in my homebrew 6800 system at BYTE, Jack Emmerichs' assembler is set for memory to memory operation without any source input or object file output peripheral operations at all. During the assembly process, the only normal IO operation is output of the listing to my Asciscope terminal at 2400 bps. This form of operation is extremely fast, and would be even faster if it were not limited much of the time by the data rate of the listing to the terminal. The speed is attained by use of large regions of memory instead of peripherals. At this writing, my 6800 is equipped with a little more than 24 K bytes of working main memory which is allocated to the software development process in the first 28 K of memory address space as follows:

0000-00FF	*Scratchpad direct addressing region used by all programs (like registers on a big machine).*
0100-0FFF	*IO Device Address Allocations.*
1000-1FFF	*Write protected programmable memory for the Interactive Manipulator Program (monitor, IO routines, text editor).*
2000-2FFF	*Assembler object file output built from 2000 upward, assembly time stack built from 2FFF downward.*
3000-5FFF	*Source string text area prepared by IMP's editor.*
6000-60F1	*Master initialization copy of directly addressed scratchpad memory contents.*
60F2-6FFF	*Tiny Assembler 6800 relocated by hand.*

In memory to memory operation, the patches include programmed simulations of the nonexistent parts of IO, as well as a memory loader program which is patched into an appropriate part of the assembler to get data prior to conversion into external form.

An Input Simulator

The normal CHRIN input routine of the MIKBUG monitor, or other similar systems software, simply returns the next character from the terminal device, which might be a Teletype, video terminal, etc. For memory to memory operation this is accomplished by use of a pointer variable and a routine which fetches the next byte and increments the pointer. If the pointer exceeds a maximum value, the assembly is terminated with an error message, so my program assumes that the end of the assembly will be signalled by an END statement prior to running out of data in the text string. End of line to the assembler is indicated by the same ASCII carriage return (hexadecimal value D) which marks the end of line for the text editor. The input simulator I use is illustrated in listing 5; this routine also assumes that initialization of the assembly sets the pointer named IPTR equal to the beginning address of the text buffer. It also is used to echo the original source listing of the assembly to the logical output devices. This always includes the 2400 bps video terminal, but might also include my Teletype when I need a hard copy listing and set an appropriate monitor flag.

The Output Simulator

Output for the character images of the object code of a line of assembled text are handled exactly as in MIKBUG using routines that accomplish the same purpose. The data goes to a terminal or Teletype hard copy device in human readable form but is not stored in machine readable form as assumed by Jack. However, I wanted to be able to load the memory region from 2000 to 2FFF with data generated by the assembler, rather than record the character format data on tape and later load it with a program such as that supplied by Jack in listing 4.

Thus using the source listing of the assembler, I identified the location in the program (the beginning of WRITE) at which all data for a given byte of memory is ready in binary form, then patched in the loader of listing 6. This loader is patched into the assembler's WRITE routine, using code shown by comments at the end of listing 6. This loader has been designed to calculate a pointer into the output buffer area with the value 2000+X where X is the present output program counter value minus the starting program counter value for the assembly. This technique works well provided that the starting program counter can be properly set. This is accomplished by setting the starting program counter to an initial value of FFFF during initialization, then making a requirement that the first statement of the assembly reference the starting address of the object code. This requirement is met by making the first statement of each assembly be an ORG statement defining the starting location of the program being assembled.

Figure 1: Bar code representation of Tiny Assembler 6800. This figure, spread over several pages, contains the complete bar code representation of Tiny Assembler 6800, version 3. The copy was sent to BYTE using Kansas City (BYTE) Standard tape format, and was processed by Walter Banks and his associates at the University of Waterloo to prepare the code in the form seen here. Table 2 was prepared from the same data using a different program for the Photon phototypesetter. The bar code text here uses variable length records in the frame format described earlier in BYTE. This format is repeated here, along with the object code format, for reference by those individuals experimenting with bar code input techniques:

Standard Format

- *Sync character, binary 10010110*
- *Frame checksum (arithmetic sum of all bytes in frame ignoring carry)*
- *Relative record identification, 8 bit ascending integer*
- *Length of frame, "n"*
- *Data of frame, total of "n" bytes, as follows:*

Specific Data Format

  - *High order address for first byte of data*
  - *Low order address for first byte of data*
  - *"n-2" bytes of data loaded beginning at address of first byte of data*

These records are absolute binary data, an image of the Tiny Assembler 6800 beginning at location 0000 and extending to location 0FFF in memory address space. An extended segment of null (hexadecimal 00) data in the table area of the assembler is skipped, so not every byte in the assembler is found in this bar code representation (the same applies to table 2).

Listing 2: The main procedure loop of Tiny Assembler 6800. The assembly begins by calling an optional subroutine, assumed to exist at the start of the symbol table region of memory, to perform user initialization. As assembled and presented in this article, the subroutine is a dummy procedure represented by an RTS (hexadecimal 39) at location 02DE. The JSR STBL-1 (hexadecimal BD 02 DE at location 7A4) is the main entry point of the assembler. It can also be patched to a permanent initialization routine outside the assembler if the user's machine has more memory than the 4 K assumed by this program.

After user initialization, the symbol table is cleared. If the symbol table was present with a transient user initialization routine starting at location 02DE, this routine is replaced by a dummy return instruction for the second and subsequent invocations of the assembler following a load of the code.

Location ENTRY is the normal "continue assembly" entry point for the program. If assembling line by line, it is possible to invoke the assembler for one or more lines, issue an END pseudooperation to terminate the assembly. Then it is possible to continue operation by returning to ENTRY from your monitor (this listing assumes Motorola's MIKBUG). Note that it is not safe to do such an END operation if there are any unresolved references, unless symbol table sorting is suppressed.

Overhead of interfacing Tiny Assembler to the Motorola MIKBUG monitor is part of the initialization that precedes the call to CMD, the main assembly procedure. After an END pseudooperation, control returns from CMD, EOJ is called to perform housekeeping operations, and MIKBUG is reentered at $E0E3 (hexadecimal).

This listing and listing 1 were produced using the PL/I cross assembler version of Tiny Assembler 6800.

Note: In order to interface this program to a nonMIKBUG monitor, the details of the stack initialization at locations 07C2 to 07C8 may need to be changed. In the earlier version 2 of Tiny Assembler used at BYTE, for example, the stack was initialized to fall at the end of the 4 K byte region in which the assembler was located. In the newer version 3, adapted to use at BYTE, we initialize the stack pointer to the end of the output buffer area of memory.

```
                    | *
                    | *
                    | *     ****************************************************************
                    | *                       MAIN PROCEDURE LOOP
                    | *
(07A4)  BD  02  DE  | MAIN    JSR   STBL-1       EXECUTE USER ROUTINES
(0001)  07  A4      |
(07A7)  CE  02  61  |         LDX   #FTBL        CLEAR THE SYMBOL TABLE AND THE
(07AA)  6F  00      | CLR1    CLR   X              FORWARD JUMP TABLE
(07AC)  08          |         INX
(07AD)  8C  07  8F  |         CPX   #STBL+1200
(07B0)  26  F8      |         BNE   CLR1
(07B2)  86  39      |         LDAA  #$39         BLOCK FURTHER
(07B4)  B7  02  DE  |         STAA  STBL-1         EXECUTION OF USER PROCS
                    | *
                    | *       THIS IS THE CONTINUE ENTRY POINT
                    | *
(07B7)  CE  00  B8  | ENTRY   LDX   #BUF1        INITIALIZE
(07BA)  DF  2A      |         STX   NXCHAR         INCOMING CHARACTER POINTER
(07BC)  7F  00  1C  |         CLR   EOJFL        CLEAR FLAGS
(07BF)  7F  00  31  |         CLR   P_POS
(07C2)  CE  07  B7  |         LDX   #ENTRY       LOAD SECONDARY ENTRY POINT FOR MIKBUG MONITOR
(07C5)  FF  A0  48  |         STX   $A048
(07C8)  8E  A0  7F  |         LDS   #$A07F       LOAD STACK POINTER FOR MIKBUG MONITOR
(07CB)  CE  00  59  |         LDX   #HEADR       WRITE OUT
(07CE)  BD  07  A1  |         JSR   STROUT         HEADING
(07D1)  BD  00  00  |         JSR   CMD          MAIN PROCEDURE LOOP - CALL PROC-COMMANDS
(07D4)  BD  00  00  |         JSR   EOJ          CALL END OF JOB
(07D7)  7E  E0  E3  |         JMP   $E0E3        JUMP TO MONITOR MAINLINE
```

Figure 1, part 1:

START HERE, SCAN DOWN PAGE

FRAME IDENTIFICATIONS

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44

Figure 1, part 2:

SCAN DOWN PAGE

FRAME IDENTIFICATIONS

45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89

Figure 1, part 3:

SCAN DOWN PAGE

FRAME IDENTIFICATIONS

90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134

Text continued from page 85

be similar from one microprocessor machine language to another even though specific instructions will differ. Like pseudocode, they can be combined into extremely complex procedures. One special structure is not shown in table 1: the code to handle external program linkages.

External linkage can be handled in two ways. One is to equate the name of an external symbol to an address and use it throughout the program. This will generate the absolute address of the external symbol

Table 3: Error messages from Tiny Assembler 6800. The format of labels, operation codes and operands of Tiny Assembler 6800 is a subset of the full Motorola Assembler specification. These error messages indicate conditions which Tiny Assembler could not handle; those conditions which are noted as "terminal" tend to propagate through the rest of the assembly generating additional errors. Error codes are single letters, listed here with some explanation.

ERROR A

Verb does not start with an alphabetic character. A verb is an operation code or pseudooperation of the assembly language. This error can result from a prior error, or from mistyping source data.

ERROR B

Verb not in table. This is the primary indication of an invalid mnemonic operation code.

ERROR C

Verb does not allow immediate addressing. An attempt was made to use immediate addressing (signified by a '#' character immediately preceding the operand) with an operation that has no immediate mode. For example,

```
ASR        #00PS
```

would generate this message.

ERROR D

Invalid verb modifier.

ERROR E

Symbol table overflow. This is a terminal error, in the sense that code following a symbol table overflow can have spurious errors. To recover, the program being assembled should be partitioned so that a smaller number of symbols is required in assembling each part. Note that the capacity of Tiny Assembler 6800 is 200 symbols in this version.

ERROR F

Label already in use. The following are several sets of duplicate labels (as interpreted by Tiny Assembler 6800):

In Label Field	In Symbol Table
THINKS THIS THIMBLES	THIS
AOUT1X AOUT2X AOUT3X	AOUX

While symbols can be arbitrarily long, the first three characters and last character must be unique.

ERROR G

Relative branch greater than 128 locations away. This error occurs if the target of a branch instruction is too far away for a signed 8 bit displacement from the present location counter value. This can usually be fixed by using a jump instruction. For example, if

```
BMI     TARG
```

gives error G, then it might replaced by the complementary condition and a jump as follows

```
BPL     DUM1
JMP     TARG
DUM1 *** continue code ***
```

ERROR H

Forward reference table full. This is a warning message that indicates imminent collapse of an assembly if another reference to an undefined symbol is found prior to the resolution of any previously unresolved reference. Tiny Assembler 6800 has a table of 25 unresolved forward references maximum.

ERROR I

Forward reference table overflow. This is the terminal message that indicates an attempt to exceed 25 unresolved forward references. After this error is noted, there will typically be numerous spurious error messages. Eliminate forward references to undefined labels to cure this problem. One way to eliminate an unresolved forward reference is to define the data or code involved prior to its first reference.

ERROR J

Invalid character in hexadecimal or decimal number. Only two types of numeric constant are allowed by Tiny Assembler 6800: up to four decimal digits, or a dollar sign ($) character followed by up to four hexadecimal digits.

ERROR K

Invalid symbol. A symbol in the label field of an operation begins with a number, or an invalid character is detected in a symbol.

for each instruction which references it. A better way is to create a table of linkages to external symbols and have all references go through the table. In this way, only one statement must be changed if the location of an external symbol should change. If all references are in the form of subroutine calls, a simple jump statement can be used to transfer control from the table to the external location. In this assembler, all calls to the monitor's IO routines go through the table shown in listing 1.

The CASE statement is somewhat more complex. The pseudocode for CASE is shown in table 1 in two forms. The pseudocode can be broken down into two functions: calculate the CASE conditions, and

Listing 3: Symbol table allocations. The $01 code at 02DE is a NOP instruction which gets replaced by a $39 to block further attempts at initialization after the assembler has cleared the symbol table. A dummy transient initialization routine of three NOPs and an RTS illustrates how such code would be entered into the assembler's source; in a real situation, one would assemble the initialization routine separately and then move the code in on top of the existing assembly code of this dummy.

```
                      | *       ************** SYMBOL TABLF
                      | *
(02DE)    01          |         FCB  $01
(02DF)                | STBL    RMB  0
                      | *
                      | *         LABEL NAME  CHAR (4)
                      | *         LABEL ADDR  CHAR (2)
                      | *
(02DF)    01          |         NOP             SPACE FOR
(02E0)    01          |         NOP             USER DEFINED
(02E1)    01          |         NOP             ROUTINES
(02E2)    39          |         RTS
(078F)                |         ORG  STBL+1200  SET FOR 200 SIX BYTE ENTRIES
(078F)    00          |         FCB  $00        TABLE STOP BYTE
```

Listing 4: An example of a loader program. This loader is based on the functional structure shown in the first part of this article, and is one of several listings accompanying this article which show how Tiny Assembler 6800 works.

```
LOCN B1 B2 B3

0000           >*****************************************
0000           >*                                       *
0000           >*     TINY ASSEMBLER DIRECTED LOADER    *
0000           >*                                       *
0000           >*****************************************
0000           >*            EXTERNAL REFERENCES        *
0000           >*****************************************
0000           >        ORG   0
0000 7E 00 00 >        JMP   CLEAR
0F79           >        ORG   $0F79
0F79           >BEGIN   EQU   *          LABEL BEGINING OF CODE
0F79 7E E0 E3 >BUG     JMP   $E0E3      MONITOR MAINLINE
0F7C 7E E0 47 >INADDR  JMP   $E047      ADDRESS IN
0F7F 7E E0 55 >INHEX   JMP   $E055      HEX BYTE IN
0F82 7E E1 AC >INCHAR  JMP   $E1AC      CHARACTER IN
0F85 7E E0 7E >PUTSTR  JMP   $E07E      PUT STRING OUT
0F88           >*****************************************
0F88           >*             GENERAL DECLARES          *
0F88           >*****************************************
000D           >CR      EQU   $D         DEFINE CONTROL
000A           >LF      EQU   $A           CHARACTERS
0004           >EOT     EQU   4
0F88 0D 0A 00 >PROMPT  FCB   CR,LF,,    DEFINE OUTPUT STRINGS
0F8B 00
0F8C 20 42 45 >        FCC   6, BEGIN
0F8F 47 49 4E
0F92 04        >        FCB   EOT
0F93 0D 0A 00 >DONE    FCB   CR,LF,,
0F96 00
0F97 20 44 4F >        FCC   5, DONE
0F9A 4E 45
0F9C 04        >        FCB   EOT
0F9D 0D 0A 00 >ERRMSG  FCB   CR,LF,,
0FA0 00
0FA1 20 42 41 >        FCC 15, BAD CHAR ERROR
0FA4 44 20 43
0FA7 48 41 52
0FAA 20 45 52
0FAD 52 4F 52
0FB0 04        >        FCB   EOT
0FB1           >
0FB1           >
0FB1           >*****************************************
0FB1           >*              START LOADER             *
0FB1           >*****************************************
0FB1 CE 00 00 >CLEAR   LDX   #0          BLANK OUT FIRST 4K
0001 0F B1
0FB4 6F 00     >LOOP    CLR   X             OF MEMORY
0FB6 8C 0F 79 >        CPX   #BEGIN        EXCEPT LOADER
0FB9 27 00     >        BEQ   GO
0FBB 08        >        INX
0FBC 20 F6     >        BRA   LOOP
0FBE CE 0F 88 >GO      LDX   #PROMPT     WRITE OUT
0FBA 03
0FC1 BD 0F 85 >        JSR   PUTSTR        PROMPT STRING
0FC4 BD 0F 82 >FNDADR  JSR   INCHAR      READ NEXT CHARACTER
0FC7 81 00     >        CMPA #0          IF END OF TAPE THEN
0FC9 27 00     >        BEQ   EOJ          GO TO END PROC
0FCB 81 28     >        CMPA #'(         IF NOT '(' THEN
0FCD 26 F5     >        BNE   FNDADR       TRY NEXT CHAR
0FCF BD 0F 7C >        JSR   INADDR      ELSE READ START ADDR INTO IX
0FD2 BD 0F 82 >NXTBYT  JSR   INCHAR      READ NEXT CHAR
0FD5 81 29     >        CMPA #')         IF END OF BLOCK THEN
0FD7 27 EB     >        BEQ   FNDADR       GO FIND NEXT BLOCK
0FD9 81 20     >        CMPA #$20        IF NOT A BLANK THEN
0FDB 26 00     >        BNE   ERROR        GO TO ERROR PROC
0FDD BD 0F 7F >        JSR   INHEX       READ NEXT CHAR INTO 'A'
0FE0 E6 00     >        LDAB X           LOAD CURRENT VALUE INTO 'B'
0FE2 1B        >        ABA              ADD 'A' AND 'B'
0FE3 A7 00     >        STAA X           STORE ANSWER IN MEMORY
0FE5 24 00     >        BCC   COUNT      IF CARRY THEN
0FE7 09        >        DEX                INCREMENT HIGH ORDER BYTE
0FE8 6C 00     >        INC   X
0FEA 08        >        INX
0FEB 08        >COUNT   INX              SET IX FOR NEXT BYTE
0FE6 04
0FEC 20 E4     >        BRA   NXTBYT     LOOK FOR NEXT CHAR IN BLOCK
0FEE           >
0FEE           >
0FEE           >*****************************************
0FEE           >*              ERROR ROUTINE            *
0FEE           >*****************************************
0FEE CE 0F 9D >ERROR   LDX   #ERRMSG    PUT OUT ERROR MSG
0FDC 11
0FF1 BD 0F 85 >        JSR   PUTSTR
0FF4 7E 0F 79 >        JMP   BUG        RETURN TO MONITOR
0FF7           >*****************************************
0FF7           >*           END OF JOB ROUTINE          *
0FF7           >*****************************************
0FF7 CE 0F 93 >EOJ     LDX   #DONE      PUT OUT FINISH MSG
0FCA 2C
0FFA BD 0F 85 >        JSR   PUTSTR
0FFD 7E 0F 79 >        JMP   BUG        RETURN TO MONITOR
1000           >*
1000           >*
1000           >        END


*** UNRESOLVED ITEMS:


*** SYMBOLS, SORT?

Y
BEGN 0F79  BUG  0F79  CLER 0FB1  COUT 0FEB  CR   000D
DONE 0F93  EOJ  0FF7  EOT  0004  ERRG 0F9D  ERRR 0FEE
FNDR 0FC4  GO   0FBE  INAR 0F7C  INCR 0F82  INHX 0F7F
LF   000A  LOOP 0FB4  NXTT 0FD2  PROT 0F88  PUTR 0F85

*
```

Listing 5: The input simulator routine. This routine, called exactly like the equivalent routine of MIKBUG, simulates an input operation by fetching the next byte of text from the text area. This byte is written to the terminal device to echo the progress of the assembly. The character string output sequence referenced on input overrun error is designed to use a string which terminates with hexadecimal FF. For users of the equivalent STROUT routine of MIKBUG, replace the last character of the message string with hexadecimal 04.

```
LOCN B1 B2 B2        CE
2000            >      ORG  $2000
2000            >* ORIGIN CHOSEN TO EASE RELOCATION BY USERS OF LISTING
2000            >*
2000            >* MEMORY TO MEMORY ASSEMBLY INPUT ROUTINE
2000            >* FOR USE WITH JACK EMMERICHS'
2000            >*          "TINY ASSEMBLER 6800" PROGRAM
2000            >*
2000            >*     WRITTEN BY CARL HELMERS CIRCA NOVEMBER 15 1976
2000            >*     ORIGINAL VERSION HAND ASSEMBLED
2000            >*     THIS VERSION EDITED AND ASSEMBLED FEBRUARY 14 1977
2000            >*        AS ILLUSTRATION FOR JACK EMMERICHS' ARTICLE
2000            >*        IN MAY 1977 BYTE...
2000            >*
2000            >* EQUATES OF VARIABLES
00F4            >TEMX EQU $F4      TEMPORARY INDEX SAVE AREA
00F2            >IPTR EQU $F2      POINTER TO INPUT TEXT AREA
2000            >*
2000            >*
2000            >*   ASSEMBLER PATCH FOR OUTPUT TO TERMINAL
FFFF            >OUTC EQU $FFFF [REPLACE WITH ACTUAL VALUE]
2000            >*
2000            >*
2000            >*   IMP6800 MONITOR EQUATES
2000            >*       FOR STRING OUTPUT PACKAGE DATA
1041            >PSTR EQU  $1041      PUT STRING ROUTINE
0020            >TXTX EQU  $20        PUT STRING POINTER INPUT ARGUMENT
00FC            >CLEA EQU  $FC        PSTR CLEAR SCREEN COMMAND
00FF            >STOP EQU  $FF        PSTR END OF STRING COMMAND
0007            >BELL EQU  $7         ASCII BELL
000D            >CR   EQU  $D         ASCII CARRIAGE RETURN
2000            >*
2000            >*
2000            >*  EXECUTABLE CODE OF NXCH "NEXT CHARACTER" ROUTINE
2000 DF F4      >NXCH STX TEMX
2002 DE F2      >      LDX IPTR
2004 A6 00      >      LDAA 0,X      A := @IPTR  [GET NEXT INPUT CHARACTER]
2006 08         >      INX           IPTR := IPTR + 1;
2007 8C 5F FF   >      CPX  #$5FFF   IF IPTR = $5FFF THEN
200A 27 00      >      BEQ  OOPS     [OOPS WE HAVE AN ERROR]
200C DF F2      >      STX  IPTR
200E 36         >      PSHA
200F 37         >      PSHB
2010 BD FF FF   >      JSR  OUTC     PRINT THE CHARACTER
2013 33         >      PULB
2014 32         >      PULA
2015 39         >      RTS           NORMAL RETURN
2016            >*
2016            >* ERROR DISASTER
2016 CE 00 00   >OOPS LDX  #OOPM     X := [ADDRESS OF MESSAGE]
200B 0A
2019 DF 20      >      STX  TXTX     TXTX := ADDRESS OF MESSAGE
201B BD 10 41   >      JSR  PSTR     PRINT MESSAGE [IMP MONITOR ROUTINE]
201E 20 FE      >GAGA BRA  GAGA      RUN AMUCK IN CONTROLLED LOOP TILL RESET
                >OOPM FCB  CLEA,BELL,CR,BELL,CR
2020 FC 07 0D
2023 07 0D
2017 20 20
2025 49 4E 50   >      FCC  22,INPUT BUFFER OVERRUN !
2028 55 54 20
202B 42 55 46
202E 46 45 52
2031 20 4F 56
2034 45 52 52
2037 55 4E 20
203A 0D
203B 07 0D 07   >      FCB  BELL,CR,BELL,CR,BELL,STOP
203E 0D 07 FF
2041            >      END


*** END - UNRESOLVED ITEMS:


*** SYMBOLS:

BELL 0007  CLEA 00FC  CR   000D  GAGA 201E  IPTR 00F2
NXCH 2000  OOPM 2020  OOPS 2016  OUTC FFFF  PSTR 1041
STOP 00FF  TEMX 00F4  TXTX 0020
```

then execute the proper case procedure. In table 1, the second form of the CASE construct and the detailed code shows this restructuring. The implementation of this structure resembles a computed GO TO, and is shown in detail in table 1. Unlike a computed GO TO, however, there is only one logical exit from the routine. This is a return from subroutine (RTS) instruction at the end of the select procedure. In the assembler, the CASE structure is used to select which procedure is to be executed for assembler directives. In our model CASE of table 1, a branch around the structure is shown for completeness; this is not necessarily required in all cases.

During the process of translating each function into assembler code, it was not difficult to follow the coding techniques developed above. Control logic is not complex because control instructions generally fall into two simple categories. First, within a given function there are almost always branch instructions which are limited in range from −128 to +127 bytes relative to the location of the next instruction. Each function is usually compact enough to fit within this range. Second, between functions control instructions are always subroutine calls. The only way for one function to affect another is through a jump to subroutine, or by changing values in program variables or machine registers. This makes the program very easy to understand and modify. *[Proven in practice here at BYTE where Jack's assembler has been undergoing a bit of customization.]*

Trying It On a Real Machine

The problem of machine access was finally solved by my friendly local Milwaukee Computer Store which donated the use of a demonstrator Southwest Technical Products M6800 for debugging the final version of the assembler. The final phase of the project could now begin.

The main line and IO modules were debugged first so that the program could communicate with the user. Functions were tested in the same top down order used in design. Once a function worked properly it was almost never necessary to make further corrections based on the testing of lower level modules. Listing 2 shows the main line procedure of the assembler, with some comments on its operation. Listing 3 shows the symbol table allocations with comments on transient user initialization performed by code in the symbol table area.

Examples of how the assembler works are shown in listings 4, 5 and 6. Listing 4 is a loader program based in part on the functional description in pseudocode developed earlier and shown in listing 5 of part 1 of this article. The style of this loader reflects its use as an example of assembler source code more than its possible use as an efficient, compact loader. Listings 5 and 6 supplied by Carl Helmers are examples of what can be done to adapt the assembly program.

Listing 6: An object file output simulator. This routine, which is patched into the assembler's WRITE routine of hexadecimal location 8B2 (68B2 in BYTE's relocated version), is used to directly load the output of the assembler into the output buffer starting at location 2000. A starting address pointer is initialized in BASE by the first address generated in the assembly, and all other addresses are calculated by subtracting the starting address pointer from the current address then adding the origin of the output buffer. The load routine results in a memory image of the final program out of the assembler, complete with all patches and fixups required by the one pass nature of the assembler. After an assembly, the MOVE routines of the monitor used at BYTE relocate the program at its intended position in memory. No loader or linkage editor is used in this system, and absolute text is all it produces; relocation of programs is done by reassembly with a different ORG value at the start of the text. [**Note** . . . This memory loader routine assumes that the output area of the assembly is cleared by initialization prior to entry into the assembler. If this is not done, change locations 203A and 203B to NOP instructions (hexadecimal 01). In either form, this routine will not calculate undefined forward reference expressions properly. For an example of the correct code for the undefined forward reference expression special case, see locations FE0 to FEB of Jack's loader program in listing 4. . . . **CH**]

```
LOCN B1 B2 B2
2000           >     ORG  $2000
2000           >* MEMORY TO MEMORY ASSEMBLY LOADER
2000           >*    MODIFICATION FOR JACK EMMERICHS' "TINY ASSEMBLER"
2000           >*    PATCHED INTO WRITE ROUTINE DUMP DATA TO
2000           >*    LOCATION 2000-2FFF FOR LATER MOVEMENT TO FINAL LOC
2000           >*    BASE MUST BE INITIALIZED TO "FFFF" PRIOR TO ENTRY
2000           >*
2000           >* WRITTEN BY CARL HELMERS,  DECEMBER 9 1976 AS UPDATE
2000           >*    TO ORIGINAL MOD CIRCA NOVEMBER 15 1976
2000           >*
2000           >* VERSION OF FEBRUARY 14 1977 ASSEMBLED FOR JACK EMMERICHS
2000           >*    MAY 1977 BYTE ARTICLE TO ILLUSTRATE ADAPTATION.
2000           >*    EQUATES TO ASSEMBLER VARIABLES ADJUSTED FOR VERSION
2000           >*    3 ASSEMBLY OF "TINY ASSEMBLER 6800"
2000           >*
2000           >*    ORG TO LOCATION $2000 TO EASE RELOCATION ...
2000           >*       ACTUAL LOCATION IS INSIDE IMP6800 MONITOR BUT
2000           >*       IT'S EASIER TO HAND RELOCATE IF LOW ORDER ZEROS
2000           >*       ARE FORCED INTO ADDRESSES
2000           >*
2000           >*    TAPE #200A LOC 188 FILE=LOADER
2000           >*
0031           >PPOS EQU $31 ASSEMBLER PRINT POSITION
00FC           >BASE EQU $FC CURRENT BASE ADDRESS, FFFF INITIALLY
2000           >*    NOTE:  ASSEMBLIES MUST START WITH "ORG" TO SET BASE
00F6           >PTR  EQU $F6 OUTPUT POINTER, 2000-2FFF RANGE
00B5           >BYT1 EQU $B5 ASSEMBLER'S OUTPUT STRING ADDRESS
00B3           >OUTA EQU $B3 ASSEMBLER'S CURRENT OUTPUT ADDRESS
00F4           >TEMX EQU $F4 TEMPORARY SAVE AREA
00FE           >ISX3 EQU $FE REALLOCATION OF ASSEMBLER VARIABLE TO AVOID
2000           >*                        MONITOR CONFLICT
2000           >*
2000 DF FE     >LDER STX  ISX3      SAVE INDEX AS IN ORIGINAL CODE
2002 DE FC     >     LDX  BASE
2004 8C FF FF  >     CPX  #$FFFF    IS BASE EQUAL TO NULL INITIALIZATION?
2007 26 00     >     BNE  NORL      IF NOT THEN GO TO NORMAL EXECUTION
2009 96 B4     >     LDAA OUTA+1    ELSE COMPUTE A BASE ADDRESS
200B 80 00     >     SUBA #0
200D 97 FD     >     STAA BASE+1         BASE = OUTA - $2000
200F 96 B3     >     LDAA OUTA
2011 82 20     >     SBCA #$20
2013 97 FC     >     STAA BASE
2015 96 B4     >NORL LDAA OUTA+1      CALCULATE OUTPUT BUFFER ADDRESS
2008 0C
2017 90 FD     >     SUBA BASE+1          PTR = OUTA - BASE
2019 97 F7     >     STAA PTR+1               = OUTA(N) - OUTA(0) + $2000
201B 96 B3     >     LDAA OUTA
201D 92 FC     >     SBCA BASE
201F 97 F6     >     STAA PTR
2021 96 F7     >     LDAA PTR+1   IS PTR LESS THAN $2FFF?
2023 80 FF     >     SUBA #$FF
2025 96 F6     >     LDAA PTR
2027 82 2F     >     SBCA #$2F
2029 24 00     >     BCC  NOLO      IF NOT THEN DO NOT WRITE DATA AT ALL
202B CE 00 B5  >     LDX  #BYT1     POINT TO ASSEMBLER OUTPUT STRING
202E C6 01     >     LDAB #1        SIMULATION OF ORIGINAL OUTPUT
2030 D1 31     >LDR1 CMPB PPOS
2032 2C 00     >     BGE  LDR2
2034 A6 00     >     LDAA 0,X
2036 DF F4     >     STX  TEMX
2038 DE F6     >     LDX  PTR
203A AB 00     >     ADDA 0,X       COMBINE NEW DATA WITH OLD
203C A7 00     >     STAA 0,X
203E 08        >     INX            POINT TO NEXT OUTPUT BYTE
203F DF F6     >     STX  PTR
2041 DE F4     >     LDX  TEMX      PREPARE FOR ITERATION
2043 08        >LDR2 INX
2033 0F
2044 5C        >     INCB
2045 C1 04     >     CMPB #4
2047 26 E7     >     BNE  LDR1
2049 CE 00 73  >NOLO LDX  #$73      SET INDEX AS IN ORIGINAL CODE
202A 1E
204C 7E 68 B7  >     JMP  $68B7     JUMP BACK TO ASSEMBLER AFTER PATCH
204F           >* THE FOLLOWING ABSOLUTE CODE IS PATCHED INTO THE
204F           >* ASSEMBLER AT ITS LOCATION STARTING AT ADDRESS 6000...
204F           >*    68B2    7E XX XX      WRITE JMP LDER
204F           >*       [NOTE XX XX IS ACTUAL ADDRESS OF LDER,
204F           >*       NOT DUMMY 2000 USED FOR THIS ASSEMBLY ONLY]
204F           >*
204F           >     END

*** END - UNRESOLVED ITEMS:


*** SYMBOLS:

BASE 00FC  BYT1 00B5  ISX3 00FE  LDER 2000  LDR1 2030
LDR2 2043  NOLO 2049  NORL 2015  OUTA 00B3  PPOS 0031
PTR  00F6  TEMX 00F4
```

Summary and Reflection

What has been presented here as a chronological listing of project activities was in fact accomplished with a fair amount of overlap. For example, a complete description of the Motorola M6800 language was not available to me until after the cross assembler was partially completed. Final table optimization was not achieved until after a version of the program was operational on the M6800 machine. Ideas came up late in development which could increase capacity or decrease overhead. In all such cases, the structured format of the program allowed modifications to be easily incorporated into existing code.

There are several possible modifications that are too major to be incorporated into the initial version of this assembler and which fall outside of the original specifications. These can be considered for future projects or may suggest still other modifications to some readers.

For example, the biggest constraint upon the use of a small assembler is the small size of the symbol table. One way to overcome this problem is to be able to delete a symbol when it is no longer needed and reuse the table space. A good way to do this is to implement a BEGIN statement. This would push the location of the most recent symbol added to the table onto a table stack. All symbols defined below this point would belong to this BEGIN block. An END statement would pull the top location off the table stack and delete all symbols below but not including this location. If there were no items in the table stack, end of program would be signaled. The nesting level of BEGIN blocks is only limited by the table stack size. This is similar to the way BEGIN and PROCEDURE structures work in PL/I and should be quite easy to implement in a sequentially searched symbol table.

If more memory were available, this assembler could be combined with an editor

to produce an assembling editor. The editor's insert, find, delete and modify functions in addition to an assembling function and a reasonably sized source file buffer would provide quite a nice program development package using about the same amount of memory required by most other assemblers.

In a large system, the assembler could be reassembled for a higher memory location and then used to load the developed object code directly into available lower memory locations. This would eliminate many of the IO timing considerations mentioned last month and the need for a loader program.

While the current package can handle large and complex programs (such as itself) and could be developed into even greater things, it is initially intended to meet more modest requirements. As a Tiny Assembler it is well suited for small programs that can be entered by hand or from fairly short input tapes on a Teletype. It is an excellent tool for the interactive development of functional blocks for a large structured program. Different methods of coding a given process can be tried to allow the user to examine the generated code. Changes can be made as errors are flagged or new ideas come up by simply backing up the program counter and recoding. (Here the loader must replace what is in memory rather than add to it, or changes will be added to errors.) Finally, if your 6800 based machine only has 4 K of memory, this is probably the only assembler that you can use at all.■

Listing 7: A multifaceted sample of various features with comments. The comments in this assembly refer to features of the assembler; the actual code is arbitrary and is not intended as a coherent program. Errors listed in table 3 are illustrated at the end of the assembly.

```
LOCN B1 B2 B2
0000           >**********************************************************
0000           >*         S A M P L E   ILLUSTRATES WITH COMMENTS        *
0000           >**********************************************************
0100           >     ORG  256              DEFAULT IS A DECIMAL NUMBER
0003           >CUNT EQU  &3               "&" INDICATES DECIMAL, TOO
0100 8E 00 00 >STRT LDS  #STAK            A FORWARD REFERENCE
0103 FE 00 00 >     LDX  ADDR             ANOTHER FORWARD REFERENCE
0106 C6 03     >     LDAB #CUNT            IMMEDIATE ADDRESSIN IF "#"
0108 96 0A     >BACK LDAA 10               DIRECT ADDRESSING IF <= 255 ($FF)
010A A1 02     >     CMPA 2,X              INDEXED ADDRESSING
010C 27 00     >     BEQ  FUND             RELATIVE (BRANCH) ADDRESSING
010E 09        >     DEX                   IMPLIED ADDRESSING
010F 5A        >     DECB                  ACCUMULATOR ADDRESSING
0110 26 F6     >     BNE  BACK
0112 A1 00     >     CMPA X                DEFAULT 0 OFFSET INDEXED ADDR.
0114 3E        >     WAI                   WAIT FOR INTERRUPT
0115 BD 00 00 >FUND JSR  SRTN             JUMP TO SUBROUTINE (CALL)
010D 07
0118           >*    [NOTE FORWARD REFERENCE IS RESOLVED HERE AND
0118           >*       PATCHED INTO EARLIER BRANCH.]
0118 B6 01 00 >     LDAA STRT             EXTENDED ADDRESSING
011B 7E 01 15 >     JMP  FUND             JUMP UNCONDITIONAL
011E 16        >SRTN TAB                   --- THESE ARE COMMENTS --
0116 01 1E
011F BA 00 00 >     ORAA BYTE             FORWARD REF, EXTENDED ADDR.
0122 39        >     RTS
0123           >     RMB  10               RESERVE 10 BYTES MEMORY
012D           >STAK RMB  1                STAK NOW RESOLVED
0101 01 2D
012E 80        >BYTE FCB  $80              FORM CONSTANT BYTE,"$" FOR HEX
0120 01 2E
012F           >*     [NOTE RESOLUTION OF EXTENDED ADDRESS,
012F           >*          AND BACKWARD PATCH...]
012F 10 00 04 > FCB $10,,4,32,$32        FORM SEVERAL CONSTANT BYTES,
0132 20 32
0134           >*    WITH NULL INDICATED BY ",,"; NOTE USE OF "$" FOR HEXA
0134           >*    DECIMAL, AND ONLY ONE SPACE AT START OF LINE .  FIRST
0134           >*    CHARACTER IS "*" FOR COMMENT, " " FOR UNLABELLED LINE
0134           >*    AND ALPHANUMERIC FOR SYMBOL DEFINITION.
0134 49 54 45 >DATA FCC 5,ITEMS          FORM 5 CHARACTER STRING
0137 4D 53
0139 01 34     >ADDR FDB DATA              FORM 2 BYTE ADDRESS
0104 01 39
013B
013B           >* NOW SOME ERRORS
013B 86 20     >     LDAA #10!!$20         USE OF "!" AS DELETE CHAR.
*** ERROR - A >LDAA 5                     LEADING BLANK MISSING
013D
*** ERROR - B >     LDA A,5               "A" SEPARATE FROM "LDA"
013D
*** ERROR - C >     STAA #DATA            IMMEDIATE ADDRESS IMPOSSIBLE
013D
*** ERROR - D >     STAA 5,W              "W" SHOULD BE "X"
013D
*** ERROR - F >DATA FCB  10               DUPLICATE LABEL (SYMBOL)
013D
*** ERROR - G >     BRA  $FFFF            BRANCH OUT OF RANGE
013D 20
*** ERROR - J >     LDAA 12FF             INVALID DECIMAL CONSTANT
013D
*** ERROR - J >ITEM EQU  $00FX            INVALID HEXADECIMAL CONSTANT
000F
00FF           >     EQU  $FF              CORRECTION TO ERROR
013D 96 00     >     LDAA
013F           >* NULL OPERAND FIELD ABOVE EQUATED TO ZERO
*** ERROR - K >     LDAA %1010            MOTOROLA'S BINARY NOT SUPPORTED
013F
*** ERROR - K >     LDAA @3               MOTOROLA'S OCTAL NOT SUPPORTED
013F
013F B6 FF F1 >     LDAA STRT+2-DATA+$23        VALID EXPRESSION
0142 B6 00 00 >     LDAA STRT/2-DATA*$23        INVALID EXPRESSION
0145           >*   [NOTE: INVALID EXPRESSION RESULTS IN BAD DATA, NO
0145           >*   MESSAGE AND SEVERAL SPURIOUS ENTRIES IN SYMBOL
0145           >*   TABLE ]
0145           >*
0145 7E 00 00 >     JMP  UNDEFINED FORWARD REFERENCE LEFT UNRESOLVED AT
0148           >     END


*** END - UNRESOLVED ITEMS:

STR2
DAT3
UNDD

*** SYMBOLS:

ADDR 0139  BACK 0108  BYTE 012E  CUNT 0003  DAT3 0000
DATA 0134  FUND 0115  ITEM 00FF  LDAA 013D  SRTN 011E
STAK 012D  STR2 0000  STRT 0100  UNDD 0000
```

Adapting Tiny Assembler 6800

The size of the source listing (over 40 pages) for Tiny Assembler 6800 prevents us from presenting the listing in the pages of BYTE. Since a source listing and more detailed user notes are items which should be available to anyone seriously interested in modifying or extending this software, we are making available a PAPERBYTES book by Jack Emmerichs, entitled *Tiny Assembler 6800*. This book contains a reprint of the two articles by Jack which have appeared in BYTE, plus additional user manual materials and the complete source listing of Tiny Assembler 6800 (along with object code in cross assembly format).

This assembler is a valuable programming aid for systems with limited memory capacity which can free you from the drudgery of machine language. The book will be available soon. For ordering information and price, watch for ads in upcoming issues of BYTE.

Last month author Mike McNatt described a number of Baudot type teleprinters that are available on the surplus market today. Let's assume that you have since rushed out in the intervening period and acquired your own Baudot teleprinter. To use it with your system, some method must be devised to enable communication between the processor and the serial Baudot device, Mike's topic in this second part of a 3 part series.

A Guide

Michael S McNatt
4658 E 57 St
Tulsa OK 74135

To use a Baudot page printer or keyboard with your computer system, some method must be devised to enable communications between the processor and the serial Baudot device. Traditionally, these interface methods have been classified as either software or hardware. Both methods seem to have supporters who are radically convinced that the method they've chosen is the best. Strictly speaking, none of the Baudot machine interface methods can be called purely software or purely hardware; they all lie somewhere in between. However, the hardware code converter obviously requires the least amount of new software generation.

Hardware Interfaces, Code Converters

The chief advantage of these converters is that they allow the use of Baudot machines with computer software which is written to input and output exclusively in ASCII code. This is a major factor when you consider that much of the microcomputer software available is written with ASCII IO. Figure 1 illustrates the use of these converters, which consist of circuitry to (a) convert the parallel ASCII code from the processor data bus into serial Baudot code for use with page printers or paper tape perforators or (b) convert the serial Baudot output from a keyboard or tape reader into parallel ASCII code for presentation to the processor data bus.

When using hardware conversion from ASCII to Baudot, a handshaking arrangement is used whereby the Baudot machine tells the processor when it is ready to accept a character. Parallel ASCII data generated by the program's output routine is then strobed into the converter via a load pulse from the processor.

The converter's ready line changes logic levels, indicating that it is in a busy state. During this state, the processor stands by while the converter

1. generates a figures or letters shift character if required,
2. converts the shift character into serial pulses which are sent to the Baudot machine,
3. converts the parallel ASCII character just loaded into an equivalent parallel Baudot character via a programmable read only memory lookup table, and
4. converts this new Baudot character into serial pulses which are sent to the Baudot machine.

The converter's ready line then changes back to its original state, indicating that it is ready to accept another character.

A variation of this approach which may not require busy and ready handshaking is to use a long, serial shift register buffer. This would in effect compensate for speed dif-

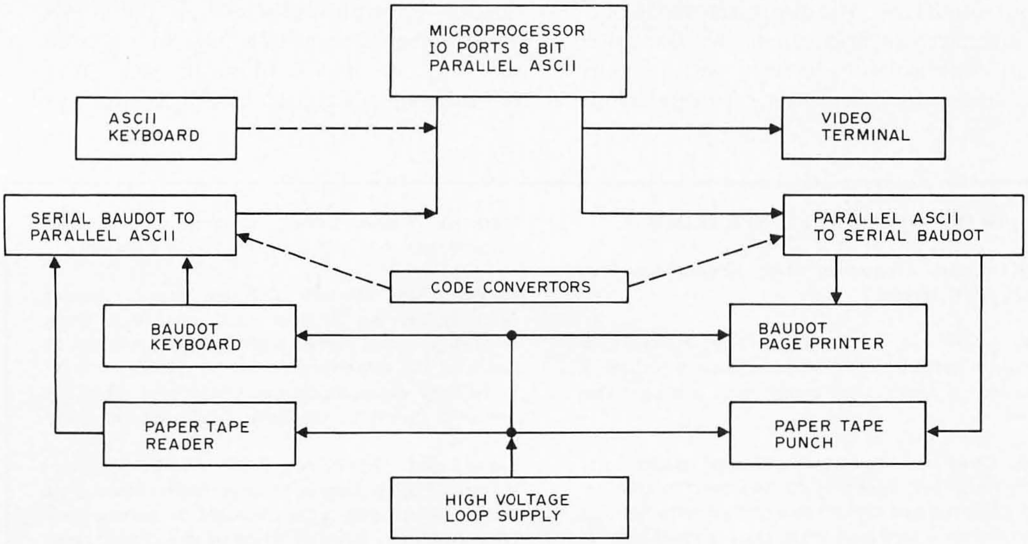


Figure 1: The block diagram describing the use of hardware code converters interfaced with a microprocessor.

to Baudot Machines:

Part 2, Interfacing Techniques

ferences in an ASCII 110 bps serial output and the slower 45 to 75 bps rates of the Baudot machines. Since most interface messages are less than one line of 88 characters in length, one line of shift register buffering may suffice; see reference 22, volume 2, #1, January 1976, page 9.

When using hardware conversion from Baudot to ASCII, serial pulses from the Baudot machine are loaded into a shift register and presented as five parallel lines representing one character to another read only memory lookup table. Circuitry in the converter keeps track of whether the character has been preceded by a letters or figures code, and presents this information to the read only memory as a sixth input line. When the parallel ASCII output from this memory is ready, the converter generates a data ready pulse, during which the processor loads in the ASCII character. This conversion operates in a manner similar to that of a standard ASCII keyboard which generates a strobe or ready pulse after each keystroke, indicating that the character may be loaded into the processor.

Circuits for accomplishing these conversions have been published, (references 11, 11a, 15, 16, and 21) and are available as printed circuit board assemblies from several of the companies listed in a separate box labelled "Sources of Baudot Hardware." References 9 and 10 present parallel to parallel code conversion circuits which require an additional UART or similar circuitry to obtain serial data for, or from, the Baudot machine.

Software Interfaces

In the purists' software approach, one routine is used to construct the serial Baudot code for output, and another routine is used to decipher the incoming Baudot code for input to the operating program. Routines used to accomplish these functions are sometimes referred to as timed software, since the bit timing of each character, 60 wpm, 100 wpm, etc, is derived from software timing loops. For example, to derive the precise 22.0 ms pulse periods required for 60 words per minute operation, a set of instructions which requires 12.7 μs to execute would be repeated 1,732 times for each bit of serial data.

The hardware requirement is held to an absolute minimum in both the input and

output interfaces. The processor serial output interface can be a simple two transistor circuit, optoisolator or relay which opens and closes the teleprinter's current loop. Several comments observed in the *RTTY Journal* (September 1973, page 13, for example) indicate that a Motorola MOC 1000 optical coupler has been used successfully up

GLOSSARY OF TELETYPE TERMS

AFSK: Audio Frequency Shift Keying; see Frequency Shift Keyer.

ASR: Automatic Send and Receive. Refers to a teleprinter system which includes a page printer, a keyboard, a paper tape punch and a paper tape reader.

Chad, Chadless: A classification of paper tape punch machines, referring to whether the chad or small paper circles remain as a part of (chadless) or are completely removed from the punched tape. A quote: "The chadless tape does not wind well, takes up more space when wound and occasionally will not read correctly if the little tabs of paper get stuck back in the holes." (Dan Nolan, *RTTY Journal,* February 1976, page 17.) However, these potential problems may be insignificant if a particular chadless machine is priced right.

Clutch: A term used to denote that current is continuously flowing in the Teletype loop supply, uninterrupted by any data input. The clutch or marking state is used to prevent the machine from running open and producing nothing but mechanical clatter.

Converter, Terminal Unit (TU), or Demodulator: A radioteletype (RTTY) term which indicates a device that accepts the two audio tones, representing mark and space, from the output of a receiver and converts them into current loop pulses for displaying characters on a page printer.

CR and LF: Carriage Return and Line Feed.

Demodulator: See Converter.

Frequency Shift Keyer, FSK: A radio Teletype term which indicates a device that converts the current loop pulses from a keyboard or paper tape reader into audio tones. An amateur radio transmitter then converts these tones to radio waves suitable for communications use.

Governor Motor: A nonsynchronous teleprinter motor, the speed of which is normally adjusted mechanically by viewing white strobe dots on the rotating assembly through a slit shutter tuning fork. Some hams have built speed controls for these machines, enabling electrical adjustment from 60 to over 100 words per minute.

KSR: Keyboard Send and Receive. Refers to a teleprinter system which includes a page printer and a keyboard only.

Ham: A radio amateur.

Loop Supply: A DC power supply used to drive teleprinter selector magnets; capable of supplying 20 or more commonly 60 mA at 130 to 260 VDC. Below 130 V, loop time constants can cause difficulty. Western Union telegraph loops usually work with 260 V.

Mark or "Closed Loop": A condition of steady current flow in the teleprinter loop; see clutch.

MARS: The Military Affiliate Radio System, whereby civilian amateur radio operators, hams, assist the armed forces with message communications, at the same time acquiring valuable training in military communications procedures. MARS is operated jointly by the Navy, Air Force and Army.

Paper Tapes: The narrow 0.375 in (0.953 cm) tape is usually gum backed and is used with a strip printer which just types messages on narrow tape. Western Union attaches strips of this tape to paper in their telegram operations. The wide 0.6875 inch (1.746 cm) tape is used with typing reperforators, chadless, and nontyping reperforators, chad type, which punch holes in the tape.

Perforator: A unit which can punch tape from an attached Baudot keyboard input only because the perforator operates from mechanical linkages to this keyboard. Cannot punch tapes from an external current interrupting source.

Reperforator: Derived from *re*ceiving *perforator.* This unit can receive serial Baudot paperpunching messages from both the keyboard and an external current loop interrupter.

RO: Read Only. Refers to a teleprinter system which includes a page printer only.

RTTY: Radioteletype: A form of amateur radio communications which uses teleprinter machines for transmitting and receiving messages via radio waves.

Space or Open Loop: A condition of no current flow in the teleprinter loop.

Stunt Box: A mechanical teleprinter attachment which performs certain nonprinting functions or stunts. These may include turning on a light, switching from nonprint to print, automatic carriage return and line feed, vertical and horizontal tabulation, backspace, two color ribbon, etc. These are rather complicated, Rube Goldberg contraptions, and are difficult to service.

Synchronous Motor: A motor which rotates synchronously at a constant speed which is directly proportional to the AC line frequency.

Terminal Unit, TU: See Converter.

Transmitter Distributor, TD, or Tape Reader: A machine which senses the characters punched into a paper tape and sends them automatically at a constant speed over a teleprinter loop circuit.

Typing Reperforator or Tape Punch: A machine which punches Baudot coded chadless holes into a paper tape and prints the associated characters right on the tape.

WPM: Words Per Minute.

to 130 VDC in this application. The serial input interface is an equally simple minimum component count circuit (see references 13 and 14).

There are two ways to input and output the serial Baudot data with this software approach:

1. Use operating software which inputs and outputs Baudot directly, as in the Baudot Monitor and Editor programs listed in reference 20.
2. Use operating software which normally uses ASCII input and output. However, provide additional code conversion lookup table software (see "Sources of Baudot Software" below) and modify the original ASCII programs to call these code conversion routines. Therefore, serial Baudot is input and output at the processor IO port. Obviously, if the programs are in read only memory and not programmable read only memory, they cannot be modified to take vectors to the software conversion routines.

Both approaches require the use of the timed software described earlier.

An In Between Interface

Another processor to Baudot machine interface method might be called a hybrid ap-

---

SOURCES OF BAUDOT SOFTWARE

1. MITS, 2540 Alamo SE, Albuquerque NM 87106. Selected programs from the Altair Software Library:

   (a) #728752 by J R Scott, $2, 36 byte program translates some Baudot character codes to ASCII equivalent.
   (b) #5-24-762 by Walter King, $2, 50 byte program loads object code from 5 level paper tape.
   (c) #5-24-765 by Walter King, $2, 71 byte program punches 5 level object tapes which can be read by #5-24-762.
   (d) #6-18-761 by James B Hansen, 215 byte program, title: ASCII to Baudot Translate Routine.
   (e) #6-22-761 by Jim Wiggins, (1) 106 byte program, title: Tape Load – Octal TLQ, (2) 186 byte program, title: Tape Dump – Octal TDQ, for Baudot Teletypes.

   The prices shown assume membership, $10 per year, in the Altair User's Club. An interesting note is that MITS is accepting no more Baudot code converter or Baudot paper tape punch and load programs into their library because it is overstocked with these types of programs.
2. M6800 User's Group Library, Motorola Inc, Microsystems, 3102 N 56th St, Phoenix AZ 85018. Reentrant ASCII to Baudot Converter Subroutine, Astrobaud. Other Baudot related programs may become available. A two year membership into this user's group is available for $100 or to anyone who submits an entry to the Program Library.
3. Intel Corp, User's Library, Microcomputer Systems, 3065 Bowers Av, Santa Clara CA 95051. Paper Tape Conversion program in 4040 language, converts 5 level paper tape information to ASCII 8 level paper tape. Other 4040 and 8080 Baudot code related programs may become available. A one year membership in the user's group is $100 or is available if a program is submitted to the library.
4. The Digital Group Inc, POB 6528, Denver CO 80206. Software Package HAM-1, Amateur Radio Ham Cassette for 8080, includes programs to send and receive RTTY Baudot, at 60, 66 and 100 wpm; $5 for cassette plus documentation.
5. See references 22a, 22b, and 23 of the bibliography.

SOURCES OF BAUDOT HARDWARE

1. Processor Technology, 2465 Fourth St, Berkeley CA 94710. 3P + S IO Module, $125 kit, $165 assembled, Altair/IMSAI plug compatible. Includes serial IO ports under software control, from 35 to 9600 bps for use with older Baudot machines.
2. HAL Communications Corp, 807 E Green St, POB 365, Urbana IL 61801. MCEM-8080 Microcomputer System, $375 assembled. Software monitor supports ASCII or Baudot code; optoisolator loop current driver and receiver.
3. Mini Micro Mart, 1618 James St, Syracuse NY 13203. Code Converter Boards: (1) Parallel ASCII to Serial Baudot kit #GPA-B, $34.95, (2) Serial Baudot to Parallel ASCII kit #TVT 2540 – BSI, $34.95.
4. MITS, 2450 Alamo SE, Albuquerque NM 87106. The Altair 680 microcomputer, $466 kit, is listed as having a five level Baudot interface option, an additional $42.
5. IMSAI, 14860 Wicks Blvd, San Leandro CA 94577. Two of the IO boards available from this company offer Baudot machine compatible serial output bps rates.
   (1) SIO 2-1, $125 kit, 75 bps minimum, compatible with 100 wmp machine such as Model 28s and property geared Kleinschmidts.
   (2) MIO, $195 kit, continuously selectable from 40 to 9600 asynchronous, compatible with 60 wpm, 45 bps, Model 15s as well as all other speeds.
6. Southwest Technical Products Co, 219 W Rhapsody, San Antonio TX 78216.
   (1) ASCII to Baudot Converter kit, $24.50, PC board, $4.35. Circuit presented in reference 15.
   (2) Baudot to ASCII Converter kit, $19.50, PC board, $4.35. Circuit presented in reference 16.
7. Polymorphic Systems, 737 S Kellogg, Goleta CA 93017. A Serial IO Option, $75 kit, can be added to the Poly-88 microcomputer kit which provides a 50 bps minimum IO rate.
8. Ohio Scientific Instruments, OSI, POB 374, Hudson OH 44236. Model 412B Supercard is designated for Baudot Terminals, $129 kit, based on MOS Technology 6502 microprocessor.
9. See reference 17.

proach, whereby parallel Baudot characters are generated by the software. These characters are presented to the data bus where busy and ready handshaking logic described in the above hardware interface description is used. In effect, the software performs the code conversion or parallel IO function and the hardware performs the parallel to serial conversion and character bit timing functions.

A UART or similar circuit (see reference 17) can be used to implement these functions. The whole procedure operates in reverse when the serial Baudot characters are input to the hardware portion of the interface and presented in parallel form to the data bus.

The Interrupt Driven Interface

These old surplus Baudot machines may of course be interfaced to the processor using an interrupt scheme, with priorities assigned to each Baudot device along with other system peripherals. Interrupt schemes are beyond the scope of this discussion since there are so many variations of software and hardware interrupt techniques, and they vary according to the processor being used. Many, and possibly most, computer hobbyist systems do not use interrupts at all. Usually in this class of system, either there are few peripherals being used or the operator has lots of time on his hands to WAIT on the peripherals, or both. One example would be a system with a keyboard and page printer combination operating with monitor and debug programs.

For a system with several IO devices and very busy software, setting up an interrupt scheme may become a matter of necessity. If

REFERENCES ON BAUDOT TELETYPES

Many of these references, especially the amateur radio magazines and books, may be available for loan from a friendly ham in your area. Some ham operators have been known to hoard all of the issues of some of their type journals, such as *Ham Radio, CQ, 73*, or *QST* magazines. It should be known that a similar affliction has been found with many BYTE readers . . .

1. *RTTY Journal*, POB 837, Royal Oak MI 48068, 10 issues per year, $.35 per issue.

2. *RTTY from A to Z* by Durward J Tucker, Corvan Publishing Corp, 1970, 214 pages, $5. Detailed information on operation, mechanical characteristics and maintenance of practically all of the older Teletype machines.

3. *The Teleprinter Handbook*, first edition by D J Goacher and J G Denny, Radio Society of Great Britain, 1973, $14.95. Distributed in US by Ham Radio Books, Greenville NH 03048. An engineering textbook covering the theory and practice of radio teleprinter equipment; 30 pages of pictures and wiring diagrams of Models 15, 19, and a few pages on the Model 28. Also covered are the design of power supplies, demodulators and Teletype test equipment.

4. *Specialized Communications Techniques for the Radio Amateur*, by the American Radio Relay League, 1975, 208 pages, $3. Chapter 5: Radioteletype, 70 pages. A condensed overview of many of the Baudot machines, a table of Teletype Gear numbers for Models 14, 15 and 28, and many other basics.

5. *The New RTTY Handbook* by Byron H Kretzman, Cowan Publishing Corp, 1962, 191 pages, $3.95. Includes a brief history of Baudot teleprinters and some equipment descriptions.

6. *RTTY Handbook* by Wayne Green, Tab Books, 1972, 320 pages, $6.95. Many basic amateur radio operating circuits such as converters, exciters and FSK circuits.

7. Shinsel, Vic: "RTTY Signal Generator," *Ham Radio Magazine,* March 1971, pages 23 to 29. RTL version of RY generator circuit.

8. Dean, J A and Rupley, J P: RCA Application Note ICAN-6267, "Astable and Monostable Oscillators Using RCA COS/MOS Digital Integrated Circuits," 8 pages.

9. *The TTL Cookbook* by Donald M Lancaster, Howard Sams Inc, pages 152 to 153. A parallel Baudot to parallel ASCII conversion circuit is presented which uses 5 ICs including 2 PROMs.

10. Ellsworth, Cole: "ASCII to Baudot Converter," *73 Magazine*, February 1976, pages 52 to 57. A 15 IC parallel ASCII to parallel Baudot code converter. Additional UART circuit required to obtain serial Baudot for teleprinters.

11. Hutton, Louis I: "Build This Exciting New TVT," *73 Magazine*, March 1976, pages 76 to 83. Included is a Teletype current loop Baudot code to parallel ASCII converter for driving a video terminal display, page 80.

11a. Hutton, Louis I: "Code Converter Using PROMs," *73 Magazine*, May 1976, pages 84 to 86. Improved circuit using readily available PROMs; refers to reference 11.

12. Andreasen, C Warren: "A Versatile TTY Generator," *73 Magazine*, April 1976, pages 90 to 91. Circuit is given that counts from 00000 to 11111 and converts to serial Baudot, thus generating a pattern of all Baudot letters and figures.

13. Whipple, Dick, and Arnold, John: "A Very Cheap IO – the Model 15," *73 Magazine*, May 1976, pages 77 to 80. Presents both software and hardware interfaces.

14. King, Walter S: "Interfacing the 60 mA Current Loop," *BYTE* magazine, August 1976, pages 96 to 97.

this is the case, information on interrupt interfacing techniques and hardware may be obtained from a variety of references, including microprocessor manufacturers' application literature or any of the excellent programming books or publications available (references 23b and 26) which deal with this subject.

Useful Miscellaneous Items

Cleaning an Old Teletype: In the course of researching for this article, I ran across an employee of a national teleprinter refurbishing house for Model 28s, 33s, and 35s, who offered some interesting suggestions for cleaning a Model 15. He said that this procedure could be followed if one obtained an extremely dirty teleprinter "fresh" out of a damp basement. The steps are as follows:

1. Thoroughly clean the machine with hot water. An ideal place for this is at a self-service car wash facility.
2. Force air dry the machine with an air gun from, for instance, one available at the auto service station with the car wash.
3. Spray all parts with a light lubricant such as WD-40.
4. Lubricate as required with oil and grease. Details of these procedures are contained in references 1 through 6. Pay particular attention to lubricating the felt separating pads, which should be saturated.

One thing I might add is to clean all electrical contacts, especially the keyboard, after the above sequence. A standard pencil eraser rubbed over the contacts until clean is an excellent way to accomplish this.

15. Smith, Roger L: "ASCII to Baudot," *Radio Electronics*, March 1976, pages 51 to 58. Corrections, April 1976, page 58, and August 1976, pages 16 and 22. Hardware code converter.

16. Smith, Roger L: "Baudot to ASCII," *Radio Electronics*, April 1976 pages 57 to 59. Hardware code converter.

17. Pasco, Richard C: "Converter Lets Processor Drive Teletypewriter," *ELECTRONICS,* October 30 1975, pages 97 to 98. Simple 6 IC TTL circuit which accepts 5 or 8 bit parallel data and converts it to serial current loop pulses at any desired bps rate.

18. *Reference Data for Radio Engineers*, fifth edition, pages 35 to 40, table 24: "Moore ARQ Code (Compared with 5 Unit Teleprinter Code)" and pages 35 to 38, table 22: "Timing of Five Unit Start Stop Teleprinter Codes."

19. Whipple, Dick, and Arnold, John: "Inexpensive Paper Tape Subsystem: Using 5 Level Tape with Computers," *73 Magazine*, July 1976, pages 145 to 148. Includes listings of paper tape read and punch software for Teletype Models 14 and 19.

20. Whipple, Dick, and Arnold, John: "A Baudot Monitor/Editor System . . . program listing for the 8080," *73 Magazine*, August 1976, pages 102 to 105. Uses the software IO described in reference 19. Also uses split octal addresses and octal data notations.

21. Guthrie, R David: "ASCII/Baudot with a PROM," *73 Magazine*, June 1976, pages 114 to 119. Article title is a misnomer – actually a serial Baudot to parallel, or serial, ASCII hardware code converter circuit is presented for any desired data rate.

22. *The Computer Hobbyist Group (North Texas) Newsletter*, 2377 Dalworth 157, Grand Prairie TX 75050.
    (a) Volume 2, #4, April 1976, pages 7 to 13. Contains listings of ASCII to Baudot and Baudot to ASCII conversion software for both the 8080 and the 6800 by Tom Gentry, James Brown, and others on the newsletter's "conversion committee."
    (b) Volume 2, #5, May 1976, pages 7 to 11. Contains a listing of ASCII to Baudot conversion software for the F8 microprocessor by Harold A Mauch.
    (c) Volume 2, #2, February 1976, "TTY Interfaces" by Harold A Mauch, pages 13 to 15.

23. *The Computer Hobbyist*, POB 295, Cary NC 27511. Back issues are $.65 each.
    (a) Volume 1, #3, January 1975, pages 19 to 21, "Interfacing a 5 Level Teleprinter" by Steve Stallings, software courtesy of Richard Smith. Contains listings of an ASCII to Baudot Translate routine, a Teletype Demo Program which generates lines of ripple pattern on the page printer, and a Baudot Teletype Output Routine which accepts Baudot characters and enables outputting in Baudot directly, if called as a subroutine, rather than converting from ASCII to Baudot.
    (b) Volume 1, #9, February 1976, pages 11 to 13, "Interfacing the Altair 8800, Part 2" by Hal Chamberlin. Includes an excellent discussion of simple single device hardware and software interrupts, vectored and polled interrupts, and an example of a keyboard interface using the interrupt technique.

24. Teletype Corp, 5555 Touhy Av, Skokie IL 60076.

25. Nettles, Walter: "Instructions for Hand Operated Typewriter Ribbon Reinking Device," reference 1, Volume 12, #6, page 16.

26. *How to Use the Nova Minicomputers*, by Data General Corp, 1971, Section 2.4 "Program Interrupt." Contains a discussion of interrupt servicing and priority and programming suggestions such as when to use interrupt.

Ribbon Renewing: When the black print ribbon begins to fade or dry up, additional ink may be applied on the top and bottom of the ribbon to provide many more miles of heavy duty use from a single roll of ribbon. Also, the printing is much darker and easier to read on both page printers and typing tape reperforators. Inks suitable for this purpose include stencil ink, stamp pad ink thinned with wood alcohol (methanol), and mimeograph or duplicator ink thinned with either naptha or turpentine; see reference 25. One ham I spoke with bragged of using 3-in-1 household oil as a thinning agent. Regardless of which type of ink is used, apply new ink sparingly until you get the knack of reinking. Allow at least 24 hours for the ink to soak into the ribbon and partially dry out before using on a printing machine.

Increasing the Operating Speed of Baudot Machines: There have been many comments in the hobbyist literature regarding the effects of increasing the speed of the older Baudot machines. Of the more popular units such as Models 14, 15, 19 and 28, only the Model 28 and some Model 14s were actually designed to operate at 100 words per minute. The other models were designed for slow speed operation and simply do not hold up well when run faster. Gordon White, the "Surplus Sidelights" columnist for *CQ* magazine, had these comments in *CQ*, August 1971, page 87:

> "It is possible to gear the old equipment up to turn 100 wpm, but particularly the Model 15 and 19 page printers do not last long at 100 speed. The Model 14 typing reperforator will do a little better, but still hops around when forced to run at 100 speed. The Model 14 transmitter distributor appears to serve adequately at 100 wpm, as does the Model 14 nontyping reperforator, if kept well oiled.
>
> "The older machines will run at 67 or 75 wpm with a gear change . . ."

The gear changes referred to are relatively simple to achieve. Faster gear sets may be obtained from most of the suppliers of Baudot teleprinters. Also, a full page table of gear set numbers required for various speeds on Model 14s, 15s, and 28s is shown on page 111 of reference 4.■

Continued from page 14

latter pending successful completion of our present negotiations).

We carry an extensive library of periodicals (including those hard to get BYTE back issues) and texts related to micro and minicomputers, hardware and software. We have also acquired the assets and inventories of B & B Book Service, plus their distribution rights.

Our associate store, Focus Scientific in Ottawa, 160 Elgin St, (613) 236-7767, began selling microcomputers in December. It is currently following the growth pattern of our Toronto outlet.

We have been guests on a financial and business show on CITY-TV, and we are going to be doing a radio talk show (The In Toronto Show) on CHUM FM, with host Larry Wilson. We will also being doing a series of television shows on the Ontario Educational Television Network with host Larry Wilson.

We are currently negotiating with groups in London, Ontario, and Montreal, Quebec, for "The Computer Place" franchises.

Stephen Pumple
The Computer Place
186 Queen St W
Toronto, Ontario
CANADA M5V 1Z1

BROADCAST AUTOMATION: DOING IT

I read with interest Mr Alvin's letter in the February issue about using an Altair 8800a for program automation. This is entirely feasible. In fact Harris Broadcast Products in Quincy IL makes an automation system based on a microprocessor. It's called System 90.

At KEZK I am implementing a commercial traffic and inventory system with an Altair 8800a. If things work out it will eventually tie in with bookkeeping, but that part is still in the future. Our system has 32 K bytes of programmable memory, two Phi-Decks with the Digital Group controller, one North Star mini disk with the Shugart drive and a PT VDM. Our printer is a DECwriter II. Most commercial business systems for broadcast are very expensive by comparison. If any other broadcast people are working on similar systems I would like to communicate with them.

Noel M Moss
Chief Engineer, KEZK Radio
1780 S Brentwood Blvd
St Louis MO 63144

BROADCAST AUTOMATION: A VOICE OF EXPERIENCE

I read with interest Mr Alvin's letter in February 1977 BYTE concerning the possible application of an Altair or similar unit for use in a broadcast automation system. As a beginning computer hobbyist and as the salesman responsible for the western area for Systems Marketing Company/Sono-Mag Corporation of Bloomington IL. I have the following comments.

SMC, the company I work for, is one of the leading builders of broadcast automation equipment. Seven years ago we brought out our first computer operand system, a hard wired ferrite core memory unit with 2048 user addresses with 24 bit words. We called it the DP-1 and are still making and selling it. It has an integral, fully programmable real time clock, LED displays, and its functions are programmed at a low level using numbers. It may also be addressed from an external Teletype (110 bps) or computer interface. Originally it sold for $16000 and now in a much improved version is $12000. This may seem like "an arm and a leg" to Mr Alvin, but the cost covers all the following:

1. The computer itself, core memory, all electronic interface, and a mammoth power supply.
2. The clock boards, their programming interface.
3. The digital switching hardware which provides a very sophisticated unit to control the broadcast magnetic tape audio units, to provide the correct responses to external human interrupts, and to provide time code generation and logging ports for the FCC required alphanumeric logging.

A second unit, the DP-2, has been introduced in the past two years. This is a 16 bit microprocessor with software and 2000, 4000, 6000 or 8000 user instructions. Included with this unit is another microprocessor controlled real time clock. Both of these units (DP-1 and DP-2) are available with video displays, digital cassette, the possibility of floppy disks, etc. The DP-2 starts at about $15000 with all the externals listed above for the DP-1.

Since we have brought these units out we have found out several things about the problem of station control:

1. The systems are very expensive to develop, requiring a lot of expensive human time. Therefore they must be able to be marketed in their original forms for several years.
2. The level of operation must be very simple and require little if any training for the operators since radio stations usually cannot pay much. Computer languages and sophisticated concepts are out.
3. Maintenance must be simple and support available since few, if any, broadcast engineers are trained in digital operations and support. Systems must be as redundant as possible and maintenance is mostly by substitution.

We find more often than not that radio stations do not supply an engineer

with the simplest scope, logic probe or meter. Therefore to a trained computer technician our equipment quite often looks childishly simple. I hope so, since that means we have accomplished our purpose.

I see no real reason (given an enormous amount of free time which you do not value) that a microcomputer advertised in BYTE could not be programmed, the digital to analog controls built and the interface to the other specialized broadcast playback gear designed and developed. It only took three and half years for us to get everything done on the DP-2 unit to get it on the market.

I don't want in any way to denigrate any ambitions to produce a new and useful controller for broadcast automation. I think that creativity of any kind is laudable and should be encouraged. But please understand the realities of what you are getting into. The old control unit you are operating with now probably cost at least $5000 twenty years ago. It is not capable of a fiftieth of what the least expensive new commercial units can do. To replace that unit with a new one amortized over five years with financing would cost about $300 to $350 per month and would be good for another 20 years.

If you decide to go ahead on your own, please let me know. Maybe I can keep you from some of the time-consuming blind alleys that we suffered through. If you don't, let me hear from you anyway.

Pete Charlton
491 Elbow Ct
Weatherford TX 76086

AH PUN, MY WORD

I would just like to say that I completely support your position on the forms of humor that Mr Kramer was objecting to, though Deborah Luhrs' pun-ch line almost reversed my opinion! I feel that to remove this humor would be to send BYTE a (very) small increment toward the level of the computer magazines over which it presides; after all, everybody needs a little pun in life.

Seriously, I congratulate you on your tremendous accomplishments in creating and improving BYTE; I wish I could think of something that needs help, but each issue seems perfect, yet they keep getting better. Keep up the good work (and the presses punning).

Richard Zulch
Assistant to the Director
Computer Science Dept
Redwood High School
Larkspur CA 94939

I always was skeptical when the authoritarian and pontifical humanities teacher I had in my high school stated the profound truth that "the pun was the lowest form of humour." How could he simultaneously say that and expound on the glories of William Shakespeare and a whole host of later, but not necessarily greater, punsters in English literature? . . . CH

MORE ON PUNS AND PUNDITS

I agree with Mr Kramer's letter in the January issue. Oh pun my word I do. Are we not safe in our 'ohms from such pundits as he criticizes. I am currently working on a plan to punish punsters before they make a hash of our memories. IC many possibilities. We should make them use odd addresses such as submarines, where the subroutine will chip away at their systems; make them queue for train buffers to ride on as Gilbert suggested; RAM them into old fashioned PROMS; load them into floppy diskos to pick up a baud paradoxies. Some even call me a punster, GO TO!

All seriousness aside, I haven't even noticed the puns Mr Kramer complains of.

I compliment you on a fine magazine. I learn something from every issue, and haven't found a really uninteresting article yet. Keep up the good work. One of these years, time and money permitting, I'll have my own microcomputer and will appreciate you even more.

Robert A Taylor
POB 638
Gate City VA 24251

KUDOS

I have been programming for many years—for the government, for private industry, on big IBM 370s and now microprocessors; but, I have never programmed for myself. I never really knew what personal computing was until I read your magazine and now I am extremely excited about the prospects.

I am very pleased with the effort your magazine is making to bring the world of software to the readers. You are building a source of reference for the established hobbyist, while your articles are providing a fine base for beginners. Hopefully, with this knowledge, people will realize the computer is not the complex social obstacle or menace to the human race, as was once thought.

Without further adieu, I welcome all you hobbyists to the wonder world of software and may it bring you all many happy days in return.

Kevin Stumpf
Proprietor
Computing Elements Development
277 Hazel St
Waterloo, Ontario N21 3P1
CANADA

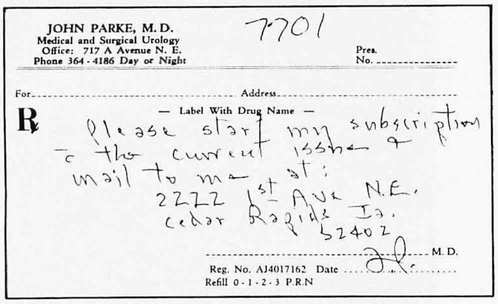
WHERE?

I really enjoyed November's BYTE. The article by Tom Buschbach called "Add This Graphics Display to Your System" is the second article in your magazine to mention Don Lancaster's

TV Typewriter Cookbook. Do you happen to know where I can purchase this book?

Glenn R Johnson
6445 SW 196th Av
Aloha OR 97005

Yes ... Order it for $9.95 from BITS. For the address see the advertisement in this issue.


JOHN PARKE, M. D.
Medical and Surgical Urology
Office: 717 A Avenue N. E.
Phone 364-4186 Day or Night
7701
Pres. No.
For ... Address ...
— Label With Drug Name —
℞ Please start my subscription c the current issue & mail to me at: 2222 1st Ave N.E. Cedar Rapids Ia. 52402
M. D.
Reg. No. AJ4017162 Date
Refill 0-1-2-3 P.R.N

A THANK YOU FOR TUTORIALS

Because I am a college student, I never find the time to finish a copy of BYTE before the next copy arrives in the mail. That causes some frustration but cannot be helped. The real frustration is not understanding many of the articles that I *do* have time to read. My knowledge of computers is limited to the programming I have done over the years coding BASIC and FORTRAN, although I do pick up some tasty bits of information from BYTE. When I find an article such as "The Types and Uses of Direct Access Storage" by Curt Hill, */page 60, January 1977, BYTE/* I positively salivate! Excellent subject; well presented. Thank you, Mr Hill!

I am sure that there are many others like me who want to understand computers before diving headfirst into highly detailed, sophisticated jargon. Please continue to keep us in mind from time to time by presenting other articles like the above mentioned one. Continue to reward potential writers out in BYTE country for their contributions. Most of all, publish! One day, when I have learned more about computers and peripherals, I will pull out my back issues of BYTE and really appreciate them.

Michael Craig
POB 895
West Acton MA 01720

On behalf of our readers, we're always looking for good tutorial articles on subjects in computer science, art and practice. Your note of appreciation for Curt Hill's article is one of the very personal profits to be derived from writing such an article.

A GOOD EXPERIENCE

In a new industry, such as the hobby computer industry, new companies always seem to pop up overnight. You keep hearing stories of problems, delays, corrections, poor documentation, etc. I even recently purchased a "burned in and tested" memory board with two of the 2102s in their sockets backwards! There are, of course, many suppliers trying hard to establish a good reputation. I would like to share my experience with one such company.

Recently I purchased an assembled CL2400 Real Time Clock Board from Comptek in La Canada CA. I called in the COD order and received it the same week. I found the board to be of high quality and professionally assembled. It was obviously tested as they even enclosed the TTY printout of the test run on my unit. The documentation was thorough including software in both basic and machine language, explanations, instructions, full troubleshooting routines, parts layout and schematics.

Now time to hook it up to my machine. The unit requires one wire to be installed from the +15 V supply to the mother board to provide the timing to the clock. It was an easy task that took less than five minutes. I then fired up my Altair and tried the program but with no luck. Then I tried the troubleshooting routines but was still stumped.

A call to California brought a return call from Ken Finster of Comptek. What evolved was two more calls with Ken doing a long distance check up of my system using me at the scope 3,000 miles away. It resulted in finding a bad 7400 that handles the ready line on my processor board. Since this was the first board that required a wait cycle it showed up the bad chip that may have been faulty from when I built my main frame.

The real time clock works perfectly now and does everything promised with ease. I offered to repay Ken for the phone bills as it wasn't even their fault, but they refused. Obviously they lost money on my sale but they still bent over backwards to make sure I was happy.

It's reassuring and a pleasure to deal with a small company such as Comptek which cares about the customer as much as the "buck." I hope that you can print this letter so that the people reading your magazine can share my good encounter. It seems the least I can do to repay Comptek for their efforts.

Gary Cooper
Vice President
S Lachman and Sons Inc
College Park MD 20740

ON TECHNOLOGICAL TRENDS AND SURPRISES

Your editorial, "Caught by Surprise," in the December 1976 issue, page 6, makes many pertinent points that some economists and certainly Ayn Rand libertarians have been pointing out for years. Modern day large technical industries as I know them I would place somewhere between zoos and three ring circuses with technical creativity and innovation the least of their concerns. Having been associated with such places for long enough that I consider my comments charitable, I always found that you end up with a corrupt, incompetent bureaucracy whose only goal is their survival — at whatever the cost. Such environments tend to filter out the better people, hound them out, or destroy them creatively and psychologically. Now, to expect the entrepreneural act to arise from such shambles is begging the ridiculous.

There have been singular studies done in the very recent past about exactly this subject. I cite: M F Wolff, "The R and D 'Bootleggers': Inventing Against Odds," *IEEE Spectrum*, July 1975, pages 38-45; A Shapero, "The Displaced, Uncomfortable Entrepreneur," *Psychology Today*, November 1975, page 83. Two recent books exist covering the same topics: R Cornuelle, *Demanaging America*, Vintage Books, New York 1975; and S L Brown, et al, *The Incredible Bread Machine*, Ward Ritchie Press, 474 S Arroyo Pkwy, Pasadena CA, October 1974.

In the broad sense, personal computing, and the activists and entrepreneurs associated with it, have reconfirmed the possibilities in a pluralistic society with at least some remnant of capitalism. To be in the midst of such events says everything is not hopeless — IBM cannot tie Digital down nor Digital the smaller guys — even though without a doubt each would like to. What we see going on now should provide hope for the individual.

Richard L Coleman
554 Springs Rd
Bedford MA 01730

REGARDING THE ANTI-TECHNOLOGIST MENTALITY

I am writing this, my first letter to an editor, in response to your December 1976 editorial, page 6. In that editorial you spoke about the growing belief that America is losing its technological edge over the rest of the world. I would like to add my own observations to yours.

At present I am a mathematics major and work full time at Stanford's Linear Accelerator Center. I have spent the last 8 years in the electronics industry. But it was not until I reenetered the university environment that I first heard people speak out against technology. One has only to sit in on any college course in the humanities to hear the growing choir of voices that rave at length against technical advance. The most commonly heard complaints are: inhuman, impersonal, unaesthetic, destructive, etc.

Unfortunately, technologists, physical scientists and mathematicians do not answer these charges, much of the time being completely unaware of them. Compounding this is the fact that their literature is not widely disseminated or inviting to the untrained. Rather, it is the very people who claim technology to be evil who are more and more getting their views widely published.

How long can American hold its lead when its young technologists are made to feel as social criminals? When everywhere they look they see themselves portrayed as dupes of a tyrannical system?

It is not against the government that we should fight, but against the growing philosophy of our age: a philosophy that calls for a return to the middle ages. It is the nature of a culture's philosophy that determines the strength of that culture. If we, the technologists, give up by default, who will be to blame?

K P Slattery
460 Auburn Way #18
San Jose CA 95129

Appropos your comment is the following quotation from Robert Heinlein's science fiction novel, The Door Into Summer.

. . . despite the crapehangers, romanticists and anti-intellectuals, the world steadily grows better because the human mind, applying itself to environment,

makes it better. With hands . . . with tools . . . with horse sense and science and engineering.

Most of these long haired belittlers can't drive a nail nor use a slide rule. I'd like to invite them into Dr Twitchell's cage and ship them back to the twelfth century – and let them enjoy it.

But I am not mad at anybody and I like now . . .

The anti-technologist mentality is an anti-people (therefore, inhuman) mentality; for the whole reason for technology in the first place is to seek new ways to improve and expand the options open to people. Hyprocrisy is rife when people use the fruits of technology while condemning the means of achieving them.

GETTING BAUD-Y

The term "baud rate" is indeed an abomination. But like "ain't," it is an assimilated abomination. And I'm not so sure it's all wrong.

Here is a line verbatim from Processor Technology's manual for the 3P+S IO module: "The baud rate can be set from 35 to 9600 baud." I suppose the writer could have said, "The baud can be set from 35 to 9600." The original version has some hope of being understood by a reader who really didn't know what "baud" meant, but the second version has little such chance.

It's too bad the term baud has become mysterious. All it means is how many times per second can you change the *state* of the signal line. If the line has two possible states, eg: current or no current, then the signaling rate in baud is the same as the information rate in bits per second. If your line has four possible states, say four different voltages, then each state change transmits two bits of information; and so in this case the signaling rate in baud is only half the information rate in bits per second. Simple, ain't it?

Jack Dennon
1157 SE 2nd
Warrenton OR 97146

No, the improved version should have been: The data rate can be set from 35 to 9600 bps; since as you point out, the two are identical for binary states.

APPALLED

I have been reading your excellent publication for a few months now, and I am appalled by the amount of obsolete TTL being used. The most flagrant example is the 7475 quad D latch (and the 74100 octal D latch). These circuits are a poor choice for almost all applications, due to their excessive input currents on both data (2 TTL loads per input) and clock (2 TTL loads per latch bit, or 4 TTL loads per 7475 clock input, or 8 TTL loads per 74100 clock input), and due to the lack of a direct clear input. The high input currents force the use of buffer gates to drive the clock inputs, which are the wrong polarity for TTL decoders anyway. The lack of a direct clear input means that peripherals such as tape drives come on flying, and must be idled by software (a direct clear input would permit a simple power on reset circuit to force the tape drives to come on idle). This is ridiculous; there are many better devices around:

1. If you don't need the $\overline{Q}$ outputs, any presettable counter can be used as a 4 bit latch to get low input currents, a direct clear input, and a load input of the proper polarity for TTL address decoders. Examples are: 74LS197, 74LS193, 74197, 74193, 74192, 74196, etc. Loadable shift registers can also be used, such as 74 95 or 74LS295. The Intel 8212 can be used, as it is an 8 bit latch intended for use as an output port. The 74LS174 or 74174 can be used to get 6 bits in a 16 pin package; the 174 behaves like the 175 discussed below, without $\overline{Q}$ outputs.
2. If you do need the $\overline{Q}$ outputs, use the 74LS175 or 74175 quad D edge triggered flip flop. This has low input currents (especially the LS version), and has a direct clear. However, the 175 lacks the transparent pass through feature of the 75. The 75, if the clock is held high, connects its inputs to its outputs, and the outputs follow the inputs as long as the clock is held high. The 175 is an edge triggered flip flop, not a latch, so the 175 outputs can change state only on the rising edge of the clock. For output port latches, the flip flop works just as well.

To repeat, use the 7475 or 74100 only if you need both the D latch (as opposed to D flip flop) function and the Q outputs, and you don't need a direct clear function.

Any TTL circuit which does not have power and ground on the corner pins should be suspected of being obsolete, as it may have been designed before the DIP package was developed, and you should look for newer circuits which may do the same job much better. Note that many circuits do jobs not hinted at by their names. For example, many presettable counters and shift registers can be used as latches. Also, multiplexers (especially the 74LS257 and 74LS258) and decoders can be used to replace logic gate arrays.

The TI book *Designing with TTL Integrated Circuits* is a good introduction to TTL, but it is five years old. Some sections are now obsolete. For example, the section on variable modulus counters was written before the advent of the 74163 fully synchronous counter. The 74163 has synchronous load and clear inputs, and can be used to make counters without the glitches and transient states inherent in counters built

around ripple counters like the 7493.

The only real cure for software theft is to make the software legally available at a price comparable to the cost of making an illicit copy, especially since hobbyists are chronically short of money.

Peter Traneus Anderson
103 N Union St
Burlington VT 05401

PS: Would you like to know some ways of expanding an 8080's instruction set? In a recent issue of *Electronics*, it is claimed that some 6800 processors will repeatedly increment their address lines, in response to a 9D op code. Are there other hidden instructions in various processors?

The new edition of The TTL Data Book for Design Engineers *is now out, and among other things it documents an impressive improvement in the range of available standard TTL parts from that source.* [It's now available from our book service, BITS Inc].

Regarding use of older parts, there is one main point worth making: I can read the new TTL data book and drool about all the new parts, but as an experimenter I have a lab inventory full of parts acquired over the years, many of which are quite serviceable if not optimal state of the art. I don't wire up ALUs with NAND gates, mind you, but if I need a latch and have a 74100 on hand but no state of the art version of this part, I'll use the 74100. An experimenter's lab is not a production shop. Eventually, the early 7400 series parts will go the way of DTL and RTL before them, but a practical attitude dictates that if the part works, it is TTL and if wiring for things like extra clock buffers is not excessive compared to a newer part, why not use it?

Regarding expansion of instruction sets, it is a neat hack to try out all the various unimplemented operation codes and see what happens. However, using such codes should be done with knowledge that they may show up in newer versions of the processor implemented in different ways. The manufacturers are not about to be bound to software and hardware compatibility with instructions they don't acknowledge to officially exist.

SEVERAL VOTES FOR "THEMES"

I don't often write letters to the editor, but I just finished reading with some interest and amusement Mr Eden's letter in the December issue. I say amusement because besides having an MSEE I have been an amateur radio operator for about 25 years and I just got through reading a similar letter from a ham in a ham magazine. The ham was complaining about the inclusion of computer articles in an amateur radio magazine and using most of the same arguments that Mr Eden used.

It should be obvious that not all computer hobbyists are hams and not all hams have any interest in hobby computers. There are, however, enough with interests in both fields to warrant inclusion of ham oriented articles in computer magazines and computer oriented articles in the ham literature.

I, for one, am not particularly display oriented at this time and so did not spend a lot of time with the November display issue. I have, however, more than made up for that issue with articles in other issues and certainly would not want to deprive those who are interested in displays of the excellent material in that issue.

To conclude, there are a lot of different interest groups in this new hobby of ours and I think you are doing an excellent job of presenting material of interest to all . . . keep it up.

Bill Hosking W7JSW
8626 E Clarendon
Scottsdale AZ 85251

AN ORIGINAL HOMEBREWER

I must be one of the original homebrewers. One day in about 1956, having read a little Norbert Weiner, and having just discovered what number base 2 was all about, I started toying with the idea of designing a digital computer just for fun. Since I knew nothing of electronics, I settled on the idea of using 4PDT relays for all logical machine elements. I felt that if I got the logical design down, it could always be implemented in more sophisticated hardware later on, on a kind of one for one substitution basis. About 6 weeks later, I found myself surrounded with literally hundreds of circuit diagrams, black box diagrams, tables of nomenclature, and a string of Fs for my term in college.

I actually built (from about 80 pinball machine relays) a crude 6 bit adder which would occasionally add two numbers correctly. My machine had a front panel that had an astonishing resemblance to an Altair 680 front panel (in 1956!). But behind the front panel, things were in pretty sad shape: total memory, two 6 bit words. Total instruction set: A+B, with some tentative provisions for A*B. System reliability: about 50% (meaning one right answer in two).

Nevertheless, it was for me an immense victory. I had, on my own, examined some of the notions essential to machine operation, such as parallel versus serial data, synchronous versus asynchronous, carry bit propagation, and so on. I had also assimilated some notion of the thousands of man years of hard work, levels of miniaturization, technical breakthroughs, and so on that lay between my machine and a practical scientific engineering desktop system. (I would guess that the average four function calculator could have been implemented in my relay hardware in a volume no larger than, say, your average

supermarket.) I resolved to stay out of the computer business until some other fellows worked out the details, and I returned to my studies for an ME.

Well, I guess those other fellows did it. We have at the office a BASIC firmware calculator/computer system (HP 9830) that exceeds my wildest 1956 dreams, and I'm spoiled rotten . . . no machine language for this kid. No card decks for overnight batch on the Corporate Monster . . . and the inevitable core dump in the morning. No more full mornings wasted trying to get the line up on the time share terminal. Sure, calculator based BASIC systems are slow in processing, but they more than make up for it in programming/debugging time and ease, plus the fact that every night of the week has 16 hours of processor time . . . *free*!

Here's my problem: Our BASIC systems (we have 2) are getting so popular it's getting very difficult to get keyboard time or even overnight time. Is it possible to use the HP 9830 as a terminal for an 8080 system (or whatever)? It seems to me that there would be a good market for a 9830/8080 interface. The HP IO goodies — cassette, keyboard, display and printer — could be used to load and unload an 8080 job, and while the 8080 is running, the HP could go about its normal duties. We could double our throughput for perhaps an additional 20% cost.

One could even envision several subsidiary 8080 processors! Further, with the output data fed back to the HP, our HP peripherals could be used (plotter, paper tape, mass memory . . . I told you I was spoiled). Any suggestions?

Ted Tharp
POB 133
Norfolk CT 06058

The only way which could probably have any semblance of being practical would be communication over a serial communications line hardwired between the two processors. The combination of these particular machines is probably not too common, and I would hardly expect HP to engineer an Altair bus interface (or vice versa). It's probably cheaper to buy another 9830.

ON HUNTING THE FORTRAN PTERODACTYL

Reading Mr Lashley's letter in February 1977 BYTE was, for me, a matter of seeing the same old wives' tale one time too many and, therefore, I feel compelled to voice my opinion on the matter. To paraphrase, I can only conclude that Mr Lashley has been theorizing too long (and practicing too little).

To call FORTRAN a pterodactyl is perhaps correct but very misleading if it is not accompanied with an explanation that FORTRAN is archaic in the sense that the English language is archaic. English is so messed up that it is impractical to program a computer to understand it directly, yet we hear very few voices advocating a wholesale switch to Esperanto. Like English, FORTRAN has become a de facto standard and unless we wish to make like the denizens of *Brave New World*, severing all ties with the past, we will have to live with it.

In his letter, Mr Lashley also states that the language one learns first influences later programming efforts. As before, I must insist on setting this statement in the proper context. There is a tendency on the part of poorly trained programmers to mistake the language for the machine. This is a deficiency on the part of the person and has nothing to do with languages, pterodactyl or otherwise. A properly educated programmer has a good understanding of what transpires in a mainframe. To achieve this understanding it is not necessary to delve into electronics or Boolean algebra, but it is essential to comprehend the concept of "a copy of the contents of box A is placed in box B" to such a degree that one can see how these simple functions can perform complex tasks.

One should begin programming in FLOWCHART, the universal symbolic language, and allow this language to become the one that influences later thoughts and actions. When, after sufficient instruction and practice, it becomes natural to describe a clearly understood sequence of machine actions with a symbolic representation of squares, diamonds and arrows, the high level language used to effect the translation becomes dependent on factors other than the programmer's personal hangups. For example, if the program at hand involves extensive matrix operations, BASIC or APL can save considerable coding time; if financial operations are involved, COBOL might be a natural choice; if universality is important, FORTRAN might be the only way to go. In any case, the programmer should never lose sight of the fact that a high level language is little more than a vehicle to get from concept to final product.

At present, supporting a full blown high level language compiler is quite an achievement for a personal computer; supporting several is out of the question. For this reason it is important to make the best possible selection and to select some obscure educational vernacular such as PASCAL because it is esthetically more pleasing, and would leave personal computing where it is right now: a lot of hardware with very little software.

Jack Crone, Systems Analyst
USC School of Medicine
Hoffman Research Ctr, Rm 805
2025 Zonal Av
Los Angeles CA 90033

BIT SLICE 360/370 MACHINES?

Is there anyone out in personal computing manufacturing land thinking of hooking up some bit slices and read only memory to make a machine which can

run IBM 360/370 instructions? For a low cost ($1000) it may not be possible to implement all the features of the architecture (virtual systems, cache memory, error checking and correction, etc), but then the hobbyist probably would not need to pay for multiple virtual systems, anyway. It should be possible to build a cheap 360 emulator to mimic a Model 30 or 40. Have you heard of any plans to do so?

Tom Koon, industrial engineering
Texasgulf Canada Ltd
POB 2002
Timmins, Ontario CANADA

No; but one acquaintance mentioned the idea of a generalized 36 bit processor board with an expandable large system's instruction set, starting at about $2000 or so.

SOME THOUGHTS ON CLUBS

On page 126 of the August BYTE you raised some very thought provoking issues regarding the functions of regional and local computer clubs. You also have asked a question which I think all computer buffs should give some consideration to: What are the pros and cons of a national home computing association and how could it relate to those of smaller groups? I discussed your editorial at a meeting of local computer buffs, Advanced Technology Research Associates (ATRA), and here are some of the ideas we brainstormed:

A. Central Issues, Problems, Trends that all computer clubs will have to handle.

1. If small system users are to realize the full benefits of the powerful yet inexpensive hardware that manufacturers are promising us, then the number of systems *and users* must increase by hundreds of thousands annually. The same applies to software when the costs of commercially produced programs can be distributed over, say, a million systems a year.
2. This growth will take place at a time when rapid technological change will be affecting not only computers, but also other devices that computers can be linked to: audiovideo systems, Electronic Funds Transfer systems and many others.
3. The uninformed people who will often not understand the *freedom* potentials of home systems will often regard computers and many computer applications with a high degree of paranoia (witness the controversies in automation and EFT). And even data processing professionals with their own systems must be able to cope with invasions of their privacy, software theft, and other new issues raised by advancing technologies. Thus, both users and nonusers can benefit if we can create environments where they can dialogue on the problems and opportunities computers present to them *personally*.
4. Items No. 1 through No. 3 seem to argue for fast growing organizations where people can exchange both facts and opinions rapidly and in great detail. Since the volume of newcomers could greatly outnumber experienced computer buffs, new methods of communication, organization, and decision making would need to evolve. The key word for rapid learning and socialization is *dialogue*.

B. Functions of Local and Regional Computer Groups

1. Obviously, many smaller computer clubs will have the effect of dispersing rapid growth more evenly. This in turn will allow traditionally organized groups to survive longer. It will also give the innovators time to try out new techniques for opening communications, such as the Homebrew Computer Club's "mapping session" which was reported in the August BYTE editorial. Eventually each group will probably develop its own unique approaches to coping with growth.
2. Club activities will give local groups valuable experience not only in computers but also in working together. Larger clubs should probably strive for a greater number and variety of projects so as to keep the work groups small. More projects also give more opportunities for people with different interests to find something that appeals to them.
3. Finally, the larger local groups will probably start experimenting with innovative methods of decision making and group organization. These may include:
   a. Revised computer assisted Roberts (Robots?) rules of order for running committee meetings with very large committees.
   b. Delphic probe issue definition techniques, handled via a conference network.
   c. Matrix analysis or decision tree charts regularly updated by computerized text editing programs. These can be used to help understand complex and constantly changing interrelated problems.
   d. Sunset rules for the table of organization and bylaws. After a given number of years the group's charter (or related portions of it) would have to be examined and approved by the group's vote. Otherwise the sunset clause would dissolve its corporate standing.

I believe that the systems analysts among local computer buffs will find many other methods of helping their organizations.

C. Functions of a National Home Computer and Small System Users Association

1. Clearinghouse for exchanging information on innovative communications, organization and decision making techniques developed in local groups. Super 8mm films, role playing, and transplanted hardware for demonstrations could be used to disperse these techniques among other groups.
2. Promote new technologies which show great potential for integrating into home computer systems. The laser video disk Martin Buchanan discussed

in the August BYTE is an excellent example of a technology that *might* be a boon for hobbyists and non-computer users alike *if* manufacturers become aware of its long range potential. Unfortunately, some observers think RCA's capacitance disk may be easier to produce. They also think the video playback unit would be less expensive due to its use of more "off the shelf" technology. And, of course, the Japanese are always getting better with video tape. Now, if a potent national group were to get into this picture by (a) giving technical advice to its favorite publication? and (b) publicizing the vast benefits the general public could reap, we could probably tip the balance. Right now people are thinking mainly of recorded movies – and most potential video disk manufacturers are busily trying to nail down distribution rights. But with some good publicity for laser disks linked to quadraphonic modular computer systems, people may not settle for less. Let's go for the moon!

3. Establish standards for new common hardware-software interfaces and ratify them by voting. Also, is ASCII always going to be the standard communication code? ASCII has 7 bits but already UARTS and USARTS can transmit 8 bit code. Eventually technological changes will have an impact on the interfaces of millions of computer users. While I'm sure BYTE will always be there to help, a national organization with elected representatives will be needed to assure fairness to all.
4. Apply political pressure, if necessary, to unreasonable restrictive practices by the phone company. Home computer technology is largely an outgrowth from telecommunications. Eventually timesharing and network techniques will combine to place tremendous computing power in the hands of the average person. But already some educational computing users are complaining about the higher costs of Ma Bell's new SMRTS charge system in California. This may be just the beginning.
5. Speaking of telecommunications, a National Home Computer Users WATTS line for relaying digital messages from one computer club to another would be a fantastic service. It would also make coordinating activities much easier. When you exceed the line's 4800 baud capacity you could start negotiating with commercial carriers.

D. Pulling It All Together: National Conferences and So On

A national conference organized from the grass roots seems a good move. But if such a conference takes place the group sponsoring it will have to be one with a great deal of organizing experience. This raises a lot of questions: Will it be a continuing elected body financed by a membership fee? Will it rotate from one computer club to another depending on who wins the bid?

I think that a volunteer group can probably do some fantastically imaginative things

with this kind of conference. It would be a great way of showing off an intraconference communications network or gaming system, for example. And, if the group organizing the event took a few hours to learn dictation techniques, they would contract with a word processing service organization to type all their correspondence. All they would have to do is pick up a telephone, dial a certain number and dictate a memo, letter or teletype to an automatic recorder. Trained stenographers would then type the message in the desired medium.

At this stage the possibilities are quite numerous; but I hope that our group has been able to help give a little perspective to the topic. Speaking for ATRA, I wish to thank you very much for launching this discussion within the pages of BYTE. You're doing a great service there. Keep up the work.

Frank Stodolka
POB 456
Minneapolis MN 55440
Attending members: Terry Belcher, David Cargo, John Stanley, Richard Tatge

BYTE'S BARQUE, IDENTIFIED

I would like to comment on the picture on page 107 of the October issue of BYTE. I am 99.44% certain the ship is the bark (barque) USCGC EAGLE. I say this since I am a 1967 graduate of the USCG Academy, and had four summer cruises aboard EAGLE as a cadet. If you get any more responses to this picture, it might be interesting to know how many other USCG Academy graduates are BYTE readers.

Clifford J Appel
6047 Woodminster Cir
Apt #12
Orangevale CA 95662 ■

BYTE's Bits

Quick and Simple Pot Input Method:

Robert Patton, W8OI, of Jester Center, Austin TX 78784, notes the publication of a very simple idea for getting input from a potentiometer with a microprocessor. The idea appeared in the March 4 1976 issue of *Electronics* magazine, page 123, and can be summarized very simply: Make the timing resistance of an NE555 type timer circuit be a potentiometer, and hook up the circuit as a triggered oneshot. Then trigger the pulse with a strobe from the computer and measure the length of the pulse with a programmed timing loop. The program watches the output of the timer through a single bit input port and terminates the loop when the end of the pulse is reached. The count obtained in the timing loop will be a function of length of the pulse as set by the variable resistance. ■

New ASCII Standards

Twenty five additional controls are being standardized for use with the American Standard Code for Information Interchange (ASCII) in 7 bit and 8 bit environments. The draft proposed American National Standard entitled *Additional Controls for Use with ASCII* is out for public comment as BSR X3.64.

The primary thrust of the standard is to encode the control functions needed by the current technology of two-dimensional character imaging input output devices, such as cathode ray tube terminals. The ANSI X3 Technical Subcommittee X3L2, Coding and Character Sets, has been working on the standard since 1969.

It is a comprehensive standard which should standardize information interchange for this decade in a manner similar to that provided by ASCII in the last decade. The standard uses a multiple character structure similar to an Escape Sequence to obtain 78 distinct functions. Furthermore, it uses a uniform method for representing both numeric and selective parameters to indicate dimensional quantities and to obtain 88 more subfunctions. Finally, 23 modes alter the interpretation of some of the functions.

The following diverse information interchange applications are addressed:

1. Interactive terminals of the cathode ray tube type
2. Interactive terminals of the printer type
3. Line printers
4. Microfilm printers
5. Software usage
6. Form filling
7. Composition imaging eg: typesetting
8. Word processing
9. Input output devices with auxiliary devices
10. Buffered and nonbuffered devices.

The types of controls include editing functions, formatting, and the specification and control of input areas, as well as certain status setting and interrogation functions, mode selection and typesetting composition functions. The standard does not contain any network or data transmission controls.

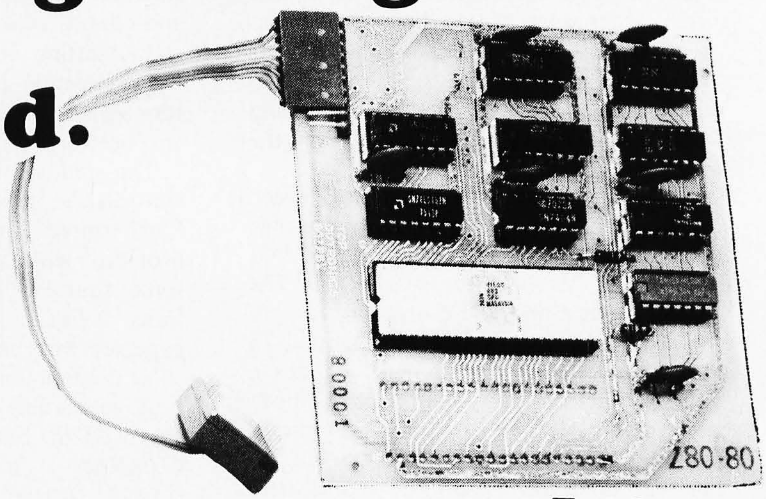
The standard is open-ended so that more controls can be included in future revisions. Furthermore, a conforming implementor is provided with a means of implementing some so-called private, nonstandard functions as well. As with ASCII, it is not expected that implementors will implement all of the functions in the standard.

A compatible subset of this dpANS was approved in June 1976 by the European Computer Manufacturers Association (ECMA). It is entitled ECMA-48 *Additional Controls for Character-imaging I/O Devices.*

Copies of the 100 page dpANS BSR X3.64 are available from R M Brown, Director of Standards, CBEMA, 1828 L St NW, Washington DC 20036. Please enclose $3 for each copy. Comments are encouraged and should be returned to CBEMA at the above address before May 28 1977.■

NOW!!

Z-80 Power for the S-100 bus without getting rid of your CPU card.



$159.95

assembled

DUTRONICS® a leader in low cost, low power ram boards has just announced it's Z 80 – 80 piggy back card. This plug – in board enables you to use your existing IMSAI, ALTAIR CPU card and upgrade your system to a Z – 80.

The card design is such that all you do is pull out your 8080 and 8212 chips, plug in the Board to the 8080 socket itself and the ribbon cable to the 8212.

A system monitor, on paper tape, is included with the board as well as a 280 Manual and Theory of Operation Manual.

Dutronics will also supply all additional software at no cost, when it becomes available

The price is $159.95 (assembled) only. OFF THE SHELF.

BYTE OF PHILADELPHIA
1345 W. Lancaster Ave.
Bryn Mawr, Penn. 19010
(215) 525–7712

BYTE OF PALO ALTO
2227 El Camino
Palo Alto, Ca. 94306
(415) 327–8080

HOBOKEN COMPUTER WORKS
20 Hudson Place
Hoboken, N.J. 07030
(201 420–1644

BYTE OF WESTMINSTER
14300 Beach Blvd.
Westminster, Ca. 92683
(714) 894–9131

BYTE OF SANTA CLARA
3400 El Camino Real
Santa Clara, Ca. 95051
(408) 249–4221

for more information call or write to:

R.H.S. MARKETING
2233 El Camino Real
Palo Alto, California 94306
(415) 321–6639

DEALER INQUIRES INVITED

B of A & MASTERCHARGE ACCEPTED

Classified Ads

FOR SALE: Two Digital Group 8 K memory boards, $180 each. Each board has 7 K 21L02-1 and 1 K 2102-1. Presently running on a Z-80 system. Send SASE and your check will be returned if boards are already sold. Tom Ringate, 2811 W 112th St, Bloomington MN 55431. (612) 881-8743.

FOR SALE: Complete wiring diagrams packet for Teletype Model 33ASR TWX with 101C Dataset, $5.50 each postpaid. Model 15 gear sets, 60, 67, 75 wpm. Gear sets for Model 28, 35 equipment. Model 28 printers, ASR, KSR, RO, sprocket feed, friction feed. Model 35 8 level ASCII TDs, reperferators, page printers. Model 35 type boxes, $40 each. Model 32/33 parts and accessories. Model 32/33 copyholders, $16 each. Write for full listing. Send SASE to Lawrence R Pfleger, 532 W Wilson St, #1, Madison WI 53703.

FOR SALE: HP-55, with recharger, battery pack, safety travel case, soft carrying case, owner's handbook, quick reference guide, programming worksheets and name labels, plus security cradle, reserve power pack (less battery) and extra recharger. Please quote offer; if no reply, outfit has been sold. Write: Frank Scavo, 1811 Court St, Syracuse NY 13208.

MICROAPPLICATIONS: I would like to correspond with anyone interested in microapplications in civil and structural engineering. David Lavers, Suite 224, 805 5th St SW, Calgary, Alberta CANADA T2P 1W3.

WANTED: Wish to buy first ten issues of BYTE. Evan H Foreman, POB F, Mobile AL 36601.

FOR SALE: Brand new Honeywell DSU-160 (IBM 2311 equivalent) 7.5 MB with full documentation, including SPC-16/40 controller diagrams, $1600. Ampex TM-7, 9 track tape transport, $450 with documentation. Roytron 75 cps paper tape punch, $350. TRW 300 cps paper tape reader, $200. Buster Killion, 2773 Winrock Av, Altadena CA 91001. (213) 798-2977.

FOR SALE: Assembled IMS UCRI cassette interface board, can also be used as 4 bit D to A and 1 bit A to D, $35. Assembled SWTPC TVT CT-L board, $12. Tarbell cassette tapes of MINOL 2.2 (a 2 K Tiny BASIC for the 8080, IO patches provided), $4. Erik Mueller, 36 Homestead Ln, Roosevelt NJ 08555. (609) 448-2605.

FOR SALE: Rockwell PPS-8 system, including CPU, IO, 4 K dynamic programmable memory with refresh and control, 256 byte scratch pad, bus interfaces, clock, and sockets. Over $400 list. Will sell for $200. Ten Signetics 300Z slices, $35. K Tyree, 2182 Coastland Av, San Jose CA 95125. (408) 265-2575 or 744-1290.

WANTED: Microdata Model Micro-400 or Micro-400/10 processor, any condition. Saul Meyer, 1734 Green Valley Rd, Havertown PA 19083.

Readers who have equipment, software or other items to buy, sell or swap should send in a clearly typed notice to that effect. To be considered for publication, an advertisement should be clearly noncommercial, typed double spaced on plain white paper, and include complete name and address information. These notices are free of charge and will be printed one time only on a space available basis. Insertions should be limited to 100 words or less. Notices can be accepted from individuals or bona fide computer users clubs only. We can engage in no correspondence on these and your confirmation of placement is appearance in an issue of BYTE.

Please note that it may take three or four months for an ad to appear in the magazine.■

FOR SALE: Innovex Model 210-M (serial #1823) floppy disk drive and one floppy disk with complete manual and schematics; Never been used. Same type of drive used in February 1977 BYTE article on interfacing a disk to an 8080 system. Will ship. $500 or best offer. Mark Becker, 32 Oxford Rd, Newton MA 02159. (617) 332-8686 after 4 PM.

FOR SALE: Processor Tech 4 K static programmable memory assembled, works perfectly, meets all specs, with all documentation, $150. Free unused paper tape of 5 K BASIC, documentation included with 4 K board. Barry Gerber, 720 Morse Av, Placentia CA 92670.

FOR SALE: 1103 1 K x 1 RAM by Intel with in house numbers. Each one individually tested with several patterns and guaranteed to meet 1103 specs. I must sacrifice 280 of these and must sell in one lot. Metal tubes are included to protect the chips during shipping. Think of what you can do with 35 K of memory! The order could also be split with a friend. You cannot beat the low price of $.80 each. I will pay shipping costs. Act quickly and send a cashiers check or money order only to Stephen Marcus, 1284 W Camino Desierto, Tucson AZ 85704.

FOR SALE: Wire wrap boards. These brand new 6 by 5½ inch boards can hold 60 14 pin ICs and can be easily modified to hold 16 to 40 pin ICs. These high density boards also have two uncommited rows to hold discrete parts. Manufactured by Augat and Cannon, they have gold plated pins and are still in their original cartons. Have several of these at $15 each. Call Bob Runge at (305) 723-6303 or write RR 2 POB 540, Melbourne FL 32901.

FOR SALE: Altair 8800 with 1 K memory. Assembled by experienced builder and in good condition. Will pay postage and handling. First money order for $425 takes it. Dave Norris, POB 2672, USAFA CO 80841.

WANTED: I need BYTE issues 3, 8 and 9 to complete my collection. Send condition of sale to Carl Shirey, 5038 Central Av, Portage IN 46368.

FOR SALE: Seals 4 K RAM board. New, assembled and run in for a 72 hour check. Features 21LO2B-1 and 8T97B buffers, complete with schematic. I'll pay US postage. Send certified check or money order for $155 to J M Church, 4307 Mar-Moor Dr, Lansing MI 48917.

FOR SALE OR TRADE: IBM 2311 disk drive, just off IBM maintenance, 7.5 MB of storage. Best offer gets it. Bill Barrow, 30 Albemarle Av, Richmond VA 23226. (804) 270-2382.

TRADE: Will trade computer programs in BASIC. I have many programs available in HP 2000 C/F and other versions. Write Bill Seurer, 4249 Browndale Av, St Louis Park MN 55416.

FOR SALE: Compucorp Beta 326 microcomputer, complete with software and TTY interface, $995. Assembled SWTPC 1024 terminal with serial interface, power supply, computer cursor control, RF modulator and case, $275. TI Sr-52, $125. Write for information to Tony Copplin, POB 3440, N Las Vegas NV 89030.

FOR SALE: MITS parallel IO board (88-PIO) for the Altair 8800, $80 assembled. Two MITS 4 K dynamic memory boards (88-4MCD) with edge connectors, $200 each, assembled. David Richards, 6655 Hill St, El Cerrito CA 94530. (415) 529-0759, evenings.

FOR SALE: Teletypes; CDC 2816 typewriter, paper tape punch and reader; CDC 9300 line printer; Mohawk Data Sciences 6021 optical card reader (Soroban ERC-1); Honeywell 404-3 Serial #BPS14 9 track, 3/4 inch tape drive; Milgo tape buffer 1807-1A Smart interface for 24 bit SDS 910; TRW graphics display system. Call Bill Palya, 120 N Weston, Rensselaer IN. (219) 866-7111.

FOR SALE: Altair 680b, assembled and running great. Best offer over $300. Alan Dahlbom, 16 Forest St, Cambridge MA. (617) 492-0676, evenings.

FOR SALE: DECwriter LA36, one year old, $1200. Brand new PDP-11/03, $1995; with additional 4 K memory, $625; total 8 K RS 232C interface, $235. List $2855, less 25% net, $2140. Also Modem Omnitek 701B, minimum six for $1000, 160 additional units. Walter D'Ull, 2239 Grand Concourse, New York NY 10453. (212) 933-0300.

FOR SALE: Floppy disks, $4 each. Computer tape 800 bpi, Ampex, Memorex, IBM, $4 a reel. Guaranteed. Trimble, 5835 Herma, San Jose CA 95123. (408) 224-0606.

FOR SALE: Altair 8800a CPU board, factory assembled and tested, never used, $250. Jonathan Rotenberg, 17 Chestnut St, Boston MA 02108. (617) 227-1399.

FOR SALE: One Altair 8800 computer, new, assembled from kit, but not tested, $439. Also, two Altair 88-4 MCD 4 K dynamic programmable memory boards, $199 each. Contact Bob Stodola, 1910 Beachwood Av, St Paul MN 55116. (612) 698-2731.

WANTED: I need six or eight 2708 PROMs (500 ns) at a reasonable price. Other pin compatible types are fine if they can be programmed in an Intel machine. Write: Theron Wierenga, 80 W Southern, Muskegon MI 49441.

FOR SALE: Issues 1 to 3 and 1 to 5 of BYTE, excellent condition, no clipped coupons. Will sell to highest bidder. Specify issues 1 to 3 or 1 to 5. K Matthews, #117 20041-55A Av, Langley BC, CANADA V3A 5V4.

WANTED: Issue #9 (May 1975) of BYTE magazine. G W Hippisley, 112 Kennedy Ln, N Syracuse NY 13212. (315) 458-0940.

TRADE: I have extra copies of BYTE issues 1 through 16 and wish to trade for INTERFACE December 1975 through June 1976. I will also buy these outright. Contact: Robert S Martin, 143 Pyne Hall, Princeton University, Princeton NJ 08540.

FOR SALE: SWTPC 6800 system, 16 K programmable memory, additional serial IO board, all ICs in new TI and Ancrona sockets. All current SWTPC software included, $725. AC-30 and Panasonic AC/DC cassette tape unit. All ICs in new Ancrona sockets, $120. All equipment was assembled with care and is in excellent condition. Louis Schubert, 3807 Greenhill Dr, Chamblee GA 30341. (404) 457-6842 or (404) 881-8800 ext 383.

FOR SALE: Viatron microprocessor system with 73 key keyboard, two cassette drives and interfaces, 9 inch video monitor and interface, power supply, cabinet and manuals, $600. Contact B Pelkey, 74 Applegate, Bennington VT 05201. (802) 447-0259.

FOR SALE: Friden Model TM20K714 keyboard-printer with TM20K15 controller and schematics. $300, you pay shipping. Also copies of schematics for above controller available for $6 to cover copying and first class shipping. Tim Ahrens, 2200 Sorret Tree Ct, Austin TX 78744.

FOR SALE: IMSAI 8080, 8 K IMSAI memory, 22 slot mother board, IMSAI audio cassette board, Processor Tech 3P+S interface, Oliver Audio Engineering OP-80A paper tape reader, plus much miscellaneous software, plugs, sockets, etc. All in like new condition, $899. Ken Schweim, 1709 Mary Ln, N Mankato MN 56001. (507) 388-5734.

WANTED: Contact with persons who have a serious personal interest in using a microcomputer for stock and commodity market investment purposes. If covariance is more than just another word to you, send a brief note of your desires, qualifications and market experience/involvement to J Williams, 2415 Ansdel Ct, Reston VA 22091. An association is being formed.

FOR SALE: Six used Augat 8170-RG1 13 slot wire wrap rack assemblies, $311 list, sell for $75 each; 16 used 8136 R series panels with 60 14 pin sockets, $15 each. Gary Manigian, POB 181, Park Ridge NJ 07656.

What's New?



Monitoring What Goes On in Your Computer

One of the most common desires of the personal computer user is the achievement of high quality output at minimum cost. Going through the RF conversion route from a video display generator to a TV set works, but is not necessarily the best way to achieve a high resolution display.

One solution to the problem is a kit of parts and information from a firm called Pickles and Trout (after its owners) which has just introduced this TV modification kit. The Pickles and Trout TVM-04 kit is available for $20 and will allow the builder to modify any inexpensive commercial television set which uses the Hitachi SX chassis, turning it into a monitor which accepts standard EIA format video signals with peak to peak amplitude levels adjustable by selection of one resistor. The Hitachi SX chassis is available in several different models of televisions including P-03, P-04, P-05, P-08, P-53 and P-63. The example shown in the photograph is a Hitachi P-05 portable which sells in typical retail and discount houses for $79.95.

As seen in the photograph, the result is a high resolution product, which will accept video generator outputs with up to 90 characters per line, more than adequate for any of the currently marketed character display modules.

How To Use It . . .

Since this product must be installed inside the commercial TV, questions of warranty and finger pointing apply. The firm has answers to such questions in the following recommendation: Purchase the TV set first, use it intensively for about a month or two of concentrated viewing (if you can stand watching TV for that long) to make sure the electronics has passed your incoming inspection. Then there is little likelihood of a manufacturing problem with the TV set, so you can safely go ahead with the changes, confident that if it doesn't work you probably made your own installation error.

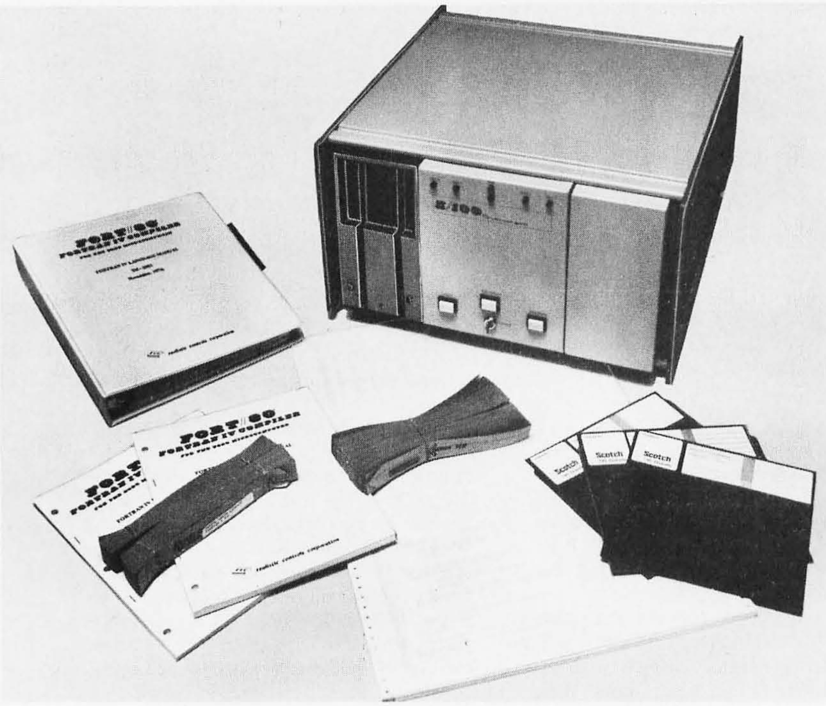
Pickles and Trout are located at POB 2276, Goleta CA 93018.■

An 8080 with FORTRAN Options

Realistic Controls Corporation has announced this new "Z//100 Series" of







tabletop personal FORTRAN machines. According to the company, the three models in the Z//100 series are portable 8080 based microcomputer systems which are available with a file management system and an optional FORTRAN IV compiler.

All three models are built around an 8080 processor. The systems software includes 3 K of PROM and a minimum of 33 K bytes of programmable memory, expandable to 64 K. The systems include an IBM compatible dual diskette drive, offering 512 K bytes of on line storage. A second dual drive can be mounted in the tabletop cabinet for a total of 1 MB of on line storage. The diskette drives feature voice coil positioning and motorized loading and unloading of diskettes.

All the Z//100 systems are available with a monitor, a complete file management system and (at extra cost) FORT//80, the company's previously announced 8080 resident FORTRAN IV compiler. FORT//80 is a subset of ANSI FORTRAN IV, producing directly executable optimized 8080 machine code. FORT//80 (including IBM format floating point) requires 11 K bytes of memory. It includes a number of language extensions which allow the user to take maximum advantage of the 8080 hardware. These include program control over interrupts and 8080 flags, direct FORTRAN addressing of 8080 ports, and the ability to link in line machine code into a FORTRAN program.

The Z//100-1 is a timesharing replacement system. Priced at $7995, it includes 36 K bytes of programmable memory, dual diskette drive and two EIA RS-232-C serial communications channels. One channel can be ordered with a 20 mA current loop interface for a Teletype. According to the company, this system is designed to allow users to merely unplug their terminal from their acoustic coupler, plug it into the Z//100-1 and begin developing application programs using FORTRAN as a high level language.

The Z//100-2, priced at $8995, includes 34 K bytes of programmable memory, dual diskette drive, and a 60 character per second bidirectional line printer (80 column friction feed standard; 132 column tractor feed optional) and keyboard.

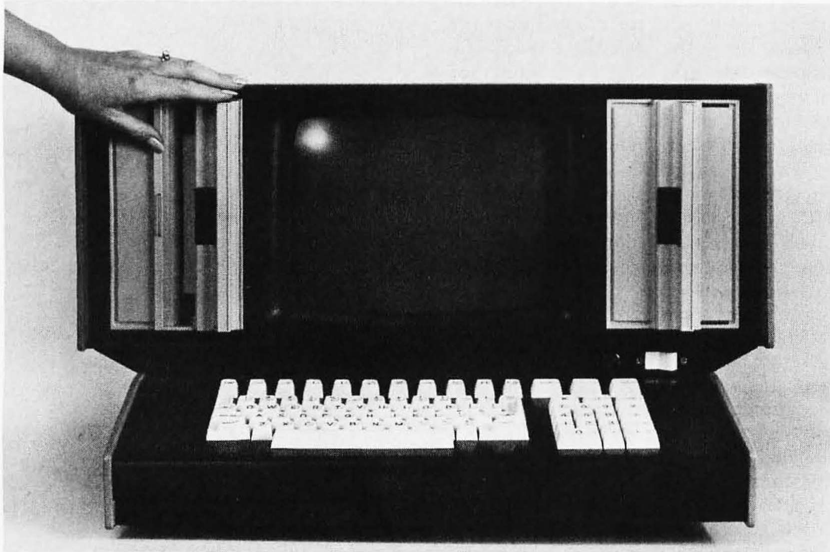
The top of the line Z//100-3, priced at $9795, includes 33 K bytes of programmable memory, dual diskette drive, a 1920 character CRT display and keyboard, and an RS-232-C serial communications channel. An optional 300 lines per minute line printer, either 80 column friction feed or 132 column tractor feed, is supported by this configuration.

The FORTRAN compiler, FORT//80, is available with any of the above Z//100 configurations, priced at $750.

Delivery is quoted as 30 days after receipt of order. Realistic Controls Corporation is located at 3530 Warrensville Center Rd, Cleveland OH 44122.■

An Advanced Intelligent Terminal System

Bitech Inc, 1440 S State College Blvd, Building 6, Anaheim CA 92806, has sent along this photo of their new brainchild: the Micro-2000 Series Distributed Data Processing System. This is a self-contained operator oriented processor which contains a video display with 24 lines of 85 characters, a standard typewriter keyboard with auxiliary numeric key pad and function keys, and dual floppy disk drives seen at either side of the display screen in this photo of the product.

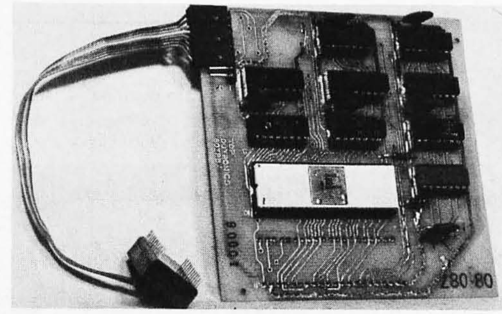


The disk controller of the Micro-2000 has an interesting feature in its use of a 16 K byte memory buffer which allows a full track of a standard size floppy disk to be transferred with a single track reference.

This processor can be ordered with main memory of from 16 K to 64 K bytes, and the press release mentions that "the Micro-2000 will support a full range of peripherals such as fixed head disks, character and line printers, 3M cartridges and 7 or 9 track tapes, as well as asynchronous communications at 75 to 9600 bps."

The Micro-2000 is said to have a complete BASIC language interpreter as well as a disk operating system which supports sequential or random access files with fixed or variable length records. An applications package oriented to business people is part of the product offerings, and delivery is quoted as 60 days.■

A Hardware Simulator That's the Real Thing

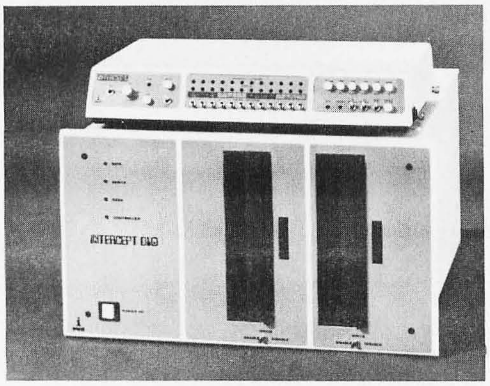


How do you get the function of a hot new processor chip without completely replacing the existing processor card of your machine? RHS Marketing, 2233 El Camino Real, Palo Alto CA 94306, has sent along this picture of a new Z-80 conversion kit for Altair, IMSAI and similar processor cards. How does it work? This card functionally duplicates the external world signals of an 8080 and its associated 8212 chip, interfacing through the sockets in which your previous 8080 and 8212 circuits had been mounted. The card is mounted "piggy back" on the original processor card, with a direct connection of the Z-80 card to the original 8080 socket, and a cable connection to the original 8212 socket. The price of this card is quoted

as $159.95 in the press release we received, and delivery is quoted as "off the shelf."■

Intercept This General Purpose System

Intersil Inc, has announced a new floppy disk program development sys-



tem, 6970-IFDOS, for use with its "Intercept" IM6100 CMOS microprocessor prototyping system. This is an industrial system which looks like a DEC PDP-8 with disk.

The purpose of the 6970-IFDOS is to provide rapid software development for systems using the IM6100 CMOS. The disk version uses the 4096 words of memory provided with the Intercept system and an ASCII terminal, such as Teletype ASR33 or equivalent.

The 6970-IFDOS consists of two interfaced floppy disk drive mechanisms with all electronics, power supplies and cables necessary to add more than 4 million bits of on line mass storage to the Intercept prototyping system.

All components are packaged in one covered enclosure which can be rack or tabletop mounted, and plugs into Intercept. It provides IBM 3740 compatible media with multiple sources, and is fully software compatible with the DEC RX8 flexible disk system for the PDP-8/E minicomputer.

The software components of the product include a file system, a keyboard monitor, a text editor, an assembler, binary loader, octal debugger, high level language interpreter, and numerous utility programs. The entire program development system is available from Intersil distributors. Price for the 6970-IFDOS floppy disk operating system is $5100. Intersil is located at 10900 N Tantau Av, Cupertino CA 95014.■

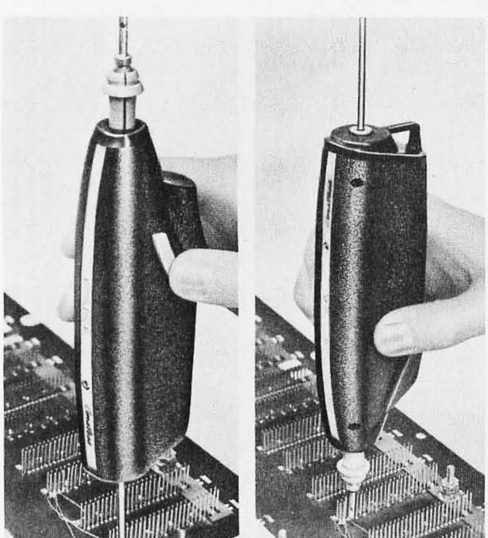
Want UV Light Sources for Erasing EROMs?



This handheld UV wand, available from Ultra-Violet Products Inc, 5100 Walnut Grove Av, San Gabriel CA 91778, is but one of the black light products available from this firm. Send for their brochure and price list on products which include various forms of lamps and stands, protective goggles, etc. This particular lamp was originally designed for use in the ultraviolet curing of certain industrial adhesives. Stores and clubs with EROM programming services will find this wand and similar products of the firm attractive for use in the erasure of previously programmed 1702A, 2704, 2708 and other EROM parts.■

What's the Inverse of a Clockwise Wrap? A Counter Clockwise Unwrap at the Other End of the Shaft

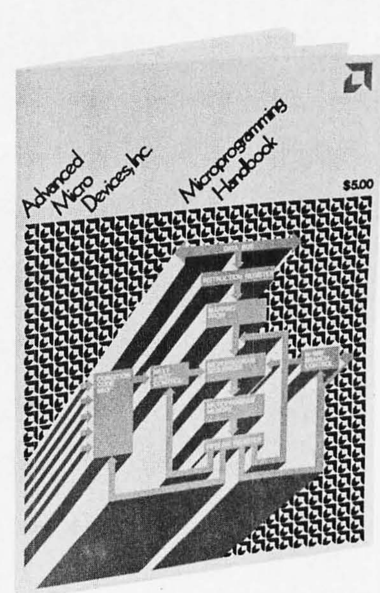
The Vector Electronic Co, 12460 Gladstone Av, Sylmar CA 91342, has just come up with this latest variation on their battery operated cordless wirewrap tool: the P160-9 wrapping and unwrapping tool, which when mounted in the P160-4R cordless motor unit gives the user both a power wrapping head (photo at right) and a power unwrapping head. This little bit of creativity costs $18.40. It can be used by itself for manual operation, or in conjunction with the P160-4R power tool which costs $45.68. It is designed to take up to three wrap levels of wire from 26 AWG to #30 AWG in size. The unwrapping end features a spring loaded sleeve that retains and ejects the wire after removal, controlling the loose ends so you don't drop said loose ends among your wire wrapped circuits causing some random logic connections or worse.■



Microprogramming – The Gory Details (Serious Homebrewers Take Note)

The Microprogramming Handbook by John R Mick and Jim Brick of Advance Micro Devices has just been made available. Its price is $5 and it is available from AMD direct (address 901 Thompson Place, Sunnyvale CA 94086) or from your friendly local distributor. Here is what it contains:

- A tutorial introduction to the Bipolar Microprocessor concept (a true microprogrammed machine as opposed to a tiny general purpose machine implemented with LSI). This introduction covers the idea of a microprogram, the concept of a control memory, organization of a control memory, branching within microsequences, overlapping instruction fetch with execution, a typical system design for a microprogrammed CPU, etc.
- A typical computer architecture implemented with the set of control chips available from AMD. Examples of the microprogram assembler available from AMD.
- Specifications of the AMD 2900 series parts.
- The high point: The complete

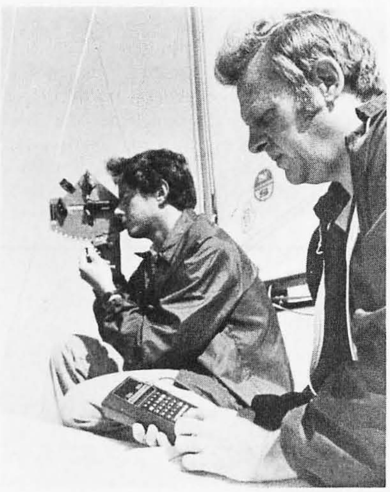


wiring diagram of a high performance microprogrammed 16 bit machine implemented with the AMD parts, contained in two D size prints. Software hackers can wire starting from this model, substitute programmable memory for the n bit by 4096 word memory shown and use your old

microprocessor system to define control memory through a "back door" and examine the results while developing your fun new computer architecture.

All in all, *The Microprogramming Handbook* will make a valuable starting point for anyone seriously interested in using the high speed contemporary bipolar chips to make a custom machine design — or an emulator of existing machines for that matter.■

Programmable Calculators at Work



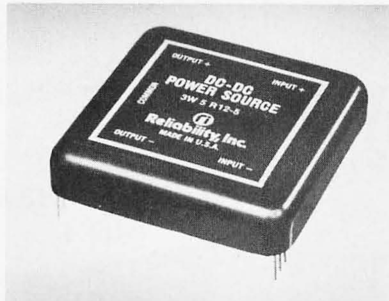
Texas Instruments supplied this picture dramatizing the use of the SR-52 programmable calculator's library of 39 navigation programs for skippers and sailors who use celestial and coastal navigation techniques, as well as ocean sailing and sailboat racing tactical strategies. The price of the programs as a package is $44.95, and the DC to DC converter needed to run the calculator in the field from mobile supplies of 12 V DC costs $12.95. Information requests should be forwarded to Texas Instruments Inc, Inquiry Answering Service, POB 5012, M/S 308, attention: SR-52 NAC PAK, Dallas TX 75222.■

Forget About Multiple Output Power Supplies for Your Homebrew 8080 . . .

The new DC to DC converter which has been designed specifically for use with microprocessors and microprocessor related integrated circuits has been announced by Reliability Inc of Houston TX.

Model 3W 5 R12-5 is a high performance power source, providing highly regulated +12 V at 200 mA and −5 V at 100 mA from a 5 VDC input. These voltages just happen to be the power supply voltages required by an 8080 processor in addition to the 5 V used by TTL circuitry. The outputs are said to be fully short circuit protected and free of damaging turn on and turn off overshoots.

The unit has low ripple and noise

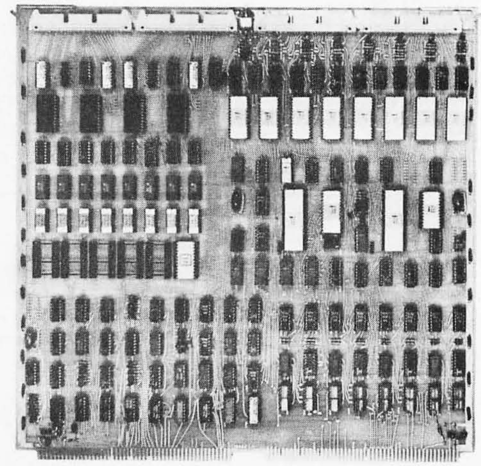


(1 mV RMS) and 0.02% line and load regulation. Input to output isolation is 300 V and operating temperature range is −25°C to +70°C. Package size is 2.0 by 2.0 by 0.40 inches. All units are encapsulated in black finished copper cases.

Availability is from stock for evaluation quantities and four weeks ARO for production quantities. For a bulletin giving full specifications and for pricing, write or call Bob Miller, Reliability Inc, 5325 Glenmont, Houston TX 77036, (713) 666-3261.■

Here's an Interesting Use of an 8080

Teknekron Inc of Berkeley CA has just introduced a new programmable interface board intended for a traditional minicomputer, the Data General NOVA



or Eclipse products. The neatness of this little concept lies in the fact that you've suddenly got a multiprocessor system when you plug one of these CPI-660A boards into the Data General product. The Teknekron interface board contains an 8080a microprocessor as its control and arithmetic element to permit customization of the behavior of the interface using a command interface discipline. Much of the detailed operation of an interface can be performed by the IO microprocessor instead of the host computer, thus freeing computing time in

the host computer for better uses like number crunching. This is hardly a personal computer product, but those readers who are involved in system design and integration may find this product of interest. Contact Teknekron at 2118 Milvia St, Berkeley CA 94704.■

Want an LSI-11 in a Desk?

Digital Equipment Corp gives you the option to get an LSI-11 with 32 K bytes of programmable memory, 512 K bytes of floppy disk mass storage on two



drives, a VT52 video terminal and an LA180 dot matrix printer. The whole system is aimed at the business person and comes with high level language software of DEC's "DIBOL" language. It's at a business person's price, too, starting at $20,900 for the configuration just described. But if you're looking for an LSI-11 development system or a complete and cheap PDP-11 system — and you have the money but not the time to configure it yourself, then this might be an attractive option. For this product, write Digital Equipment Corp's Commercial Products Group (Maynard MA 01754) and ask for information on the 320 Series of DEC Datasystems.■

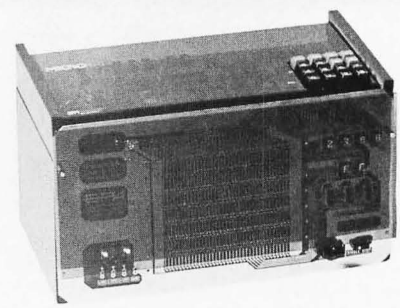


Here's a Neat Concept — For Hard Surface Disk Users With Butterfingers

The "Shockwatch" is an impact recorder for use on front load disk cartridges. It will change in appearance from clear to brilliant red if subjected to a mechanical shock or force great enough to cause a malfunction of the cartridge. Instructions are clearly shown on the label directing the operator to refrain from using a questionable cartridge. According to the press release, the Burroughs Corporation has adopted the Shockwatch for use on all B9385-3 disk cartridges.

It is intended to eliminate down time and prevent head crashes, job reruns, data loss, and the general reduction in operating efficiency which accompanies these occurrences. Information is available from Media Recovery Inc, Box 867, Graham TX 76046.■

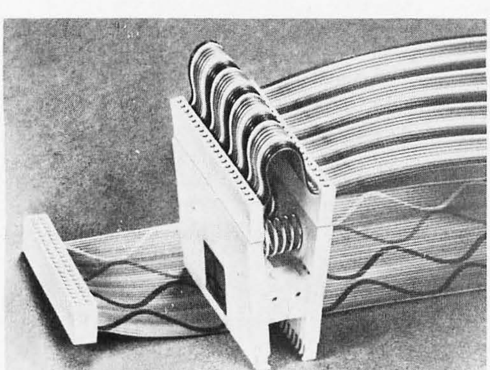
Here's an Interesting Front Panel Concept



For those interested in the details of single step operation of software, Electronic Products Associates Inc has just announced a new feature, called the "Data Catcher," to their Micro-68 line of prototype development systems. This feature will enable the user to capture the address and operand after the completion of each machine instruction and display the data on its integral 6 digit hexadecimal display. The option is priced at $140 and works with the company's "Expanded —68" microcomputer system. This kind of operation is extremely valuable in educational and learning situations, as well as when debugging or checking out small segments of production program code which must be highly optimized. For further information, contact Electronic Product Associates Inc at 1157 Vega St, San Diego CA 92110.■

Here's an Umbilical Cord for Your Microprocessor . . .

AP Products, Box 110, 72 Corwin Dr, Painesville OH 44077, has just introduced a new Test Clip Jumper Assembly which is used in conjunction with their 40 pin dual in line package test clip as



shown in this photograph. The jumper connects the test clip to a test system via a 40 wire flat cable. AP gives several termination options to the test equipment manufacturer who would use this item: no termination, double row 0.1 inch (0.254 cm) matrix socket connector, card edge connector, printed circuit connector, dual in line plug connector, etc.

This product's intended use is as the umbilical cord connecting an automatic testing device to a 40 pin package in the system under test. The test clip is connected directly to the package in the system under test, and the contacts on the test clip are routed to the corresponding wires on the cable. The test clip jumper and 12 inches (30.5 cm) of ribbon cable ending in a 40 pin double row 0.1 inch matrix socket connector as shown here cost $13.88 in quantities of 10; to this must be added the TC-40 test clip itself which costs $21 in quantities of 10. AP will cut the ribbon cable optionally to any length desired, rounded to the nearest half inch increment, presumably with a variation in price which is a function of length.■

The OPUS/ONE Language . . . Continued

Administrative Systems Inc, 222 Milwaukee, Suite 102, Denver CO 80206, has just forwarded us a copy of the *User's Manual* for the OPUS/ONE language. This manual accompanied a press release which updates the terms of the product offering (see page 96 of December 1976 BYTE) and announces three Altair compatible versions of the language. This interpretive compiler product gives the user a high level language facility which is comparable in many respects to ALGOL or FORTRAN, yet retains the simplicity of interactive BASIC. The new Altair compatible versions are expected to be released in the first quarter of 1977 (about the time you're reading this), and the price of the language has been dropped to $99 — with refunds going to all previous purchasers who paid $250 or $300 which was the previously announced price. It is available in the following three MITS Altair compatible forms:

```
                    V.  SAMPLE SORT PROGRAM

FINE

LIST 0,0,0,3

10    REM 'EXAMPLE OF BUBBLE SORT'
20    DIM M( 50) ;
30    LOOP I, 1TO 50; M( I) = RND ; NEXT ;
40    LIN 2; PRINT 'ORIGINAL NUMBERS:'; LIN 1;
50    GOSUB 'PRINT MATRIX';
60    LOOP I, 1TO 49; LOOP J, I+ 1TO 50;
70    IF M( J) < M( I) [ X= M( I) = M( J) = X] ;
80    NEXT ; NEXT ;
90    LIN 2; PRINT 'SORTED NUMBERS:'; LIN 1;
100    GOSUB 'PRINT MATRIX';
110    END ;
120    'PRINT MATRIX';
130    LOOP I, 1TO 5; LOOP J, 1TO 10;
140    FF 'R1.4B', M( 10* ( I- 1) + J) ;
150    NEXT ; PRINT ; NEXT ;
160    RETURN ;

FINE

COM

FINE

RUN

ORIGINAL NUMBERS:

 .1845  .8271  .6272  .3710  .2230  .3698  .4621  .1652  .8816  .4584
 .4405  .1849  .8947  .3897  .1719  .1021  .7467  .2107  .0156  .3386
 .6720  .7182  .0810  .0663  .6548  .4591  .2274  .2071  .4297  .7303
 .0643  .4329  .4809  .5786  .9509  .0164  .4738  .1970  .3200  .8391
 .8202  .1520  .1600  .1413  .6343  .9849  .3522  .4612  .7977  .6881

SORTED NUMBERS:

 .0156  .0164  .0643  .0663  .0810  .1021  .1413  .1520  .1600  .1652
 .1719  .1845  .1849  .1970  .2071  .2107  .2230  .2274  .3200  .3386
 .3522  .3698  .3710  .3897  .4297  .4329  .4405  .4584  .4591  .4612
 .4621  .4738  .4809  .5786  .6272  .6343  .6548  .6720  .6881  .7182
 .7303  .7467  .7977  .8202  .8271  .8391  .8816  .8947  .9509  .9849

FINE
```

- Paper tape form, loaded using a MITS 2SIO board at ports 0 and 1.
- Audio cassette form, loaded using a MITS ACR board at ports 6 and 7.
- Floppy disk form, loaded using the MITS controller at ports 10, 11 and 12.

All three versions use a MITS 2SIO board or 4PIO board interfaced to an ASCII terminal to accomplish initialization using an interactive sequence. Memory requirements are 8 K for the paper tape and audio cassette forms, and 12 K for the floppy disk version.

Whichever form you use, you get a language which features multiple precision arithmetic up to 126 decimal digits, strings both as procedure names and as a data type, variable names of unrestricted length, up to 255 dimensions to arrays [*but who has got enough memory address space to use them all?*], numerical or string data types, formatted IO statements and a block structure similar to the ALGOL BEGIN . . . END block. To give you an enticing sample of what OPUS/ONE is like, we've reproduced the sample program from page 85 of the OPUS/ONE *User's Manual* to illustrate its features.■

BYTE's Bits

Don't Miss the MACC Computerfest 77

The following notice was received from MACC as this issue was going to press . . .

A large flea market will be among the attractions at Computerfest 77, the second annual Midwest Regional Computer Conference for hobbyist computing. Sponsored by the Midwest Affiliation of Computer Clubs, an organization of amateur computing clubs in ten

neighboring states and Canada, the convention will open on June 10 at the Bond Court Hotel in Cleveland OH. Computerfest 77 will be a complete convention with workshops, manufacturers' exhibits, a flea market, seminars and technical sessions open to all attendees.

MACC is a volunteer, nonprofit organization. "Our intention is simply to bring together the buyers and the sellers, the hobbyists and the manufacturers, and have some fun doing it," says Gary Coleman, president of MACC. The ticket price is $2 for all activities, including seminars. MACC is also working on chartering a plane flight to Dallas for the NCC show's activities the week following the Computerfest. Additional details on Computerfest 77 and the NCC charter flight are available from Gary Coleman at the Midwest Affiliation of Computer Clubs, POB 83, Cleveland OH 44141.■

BYTE's Bugs

Errors in the MIKBUG Roadmap

Thanks to Douglas B Brumm PhD, 224 Hecla St, Lake Linden MI 49945, for pointing out some glitches in the MIKBUG roadmap article, page 96, in the February 1977 issue. The following are his findings:

- Table 1, page 98, OUT2H entry: The text should read "Hex byte output: The data at the location pointed to by the X register is sent to the terminal as two hexadecimal digits."
- The entry point of the print data routine is PDATA1. The PDATA2 entry should be changed to PDATA1 with address E07E in table 1.■

The Case of the Missing Trigonometry . . .

Due to an oversight of coordination, the equations accompanying figure 2 of Robert Grappel's article on page 34 of March 1977 BYTE were omitted. The trigonometry of the model aircraft servo plotter is as follows:

Equations (R = arm length)

$$(X, Y) = ((X_\theta + X_\phi), (Y_\theta + Y_\phi))$$
$$X_\theta = R \cos(\theta - 90^\circ) = -R \sin(\theta)$$
$$X_\phi = R \cos(\phi + \theta - 90^\circ) = -R \sin(\phi + \theta)$$
$$Y_\theta = R \sin(\theta - 90^\circ) = -R \cos(\theta)$$
$$Y_\phi = R \sin(\phi + \theta - 90^\circ) = R \cos(\phi + \theta)$$

Applying the above gives

$$X = -R(\sin(\theta) + \sin(\theta + \phi))$$
$$Y = R(-\cos(\theta) + \cos(\theta + \phi))$$

There is no unique solution for (θ, ϕ) coordinates except at extremes. ■

SOME NOTES ON ANALYSIS OF PLOT ANGLES

The following letter is self-explanatory and concerns inverting the calculation shown above.

I am a recent subscriber, and have now read three issues of your magazine. I have found your articles both informative and practical. I have been involved with computers for 16 years and have observed a wide range of abilities and attitudes among people in the field. The somewhat naive approach frequently taken by computer hobbyists often enables them to succeed where computer scientists with more classical approaches have failed.

With reference to figure 2 on page 34, March 1977, you indicate the need for a reiterative calculation to solve for theta and phi for your servo plotter. Here, you are guilty of assuming that there is no direct solution when there really is one. The solution may take many forms, but assuming an arcsine function is available, which produces results in degrees from -90 to +90, the following FORTRAN-like statements will be well-behaved except at the origin:

```
RHO=SQRT (X*X+Y*Y)
PSI=ASIN (Y/RHO)
XI=ASIN (RHO/ (2.0*R) )
PHI=180.0-2.0*XI
THETA=PSI+XI
```

Note that 2.0*R may be entered as a constant or precomputed once in the program. Also note that standard FORTRAN would require converting the arcsines from radians to degrees by multiplying by the constant 180.0/pi. You might also want to leave all the angles in radians, in which case the 180.0 would be changed to pi, since the angles will be rescaled before being output to the servos anyway.

Theta and phi should be range checked to avoid overdriving the servos. There may also be some mechanical interference to avoid, depending on the design. These interference problems may be eliminated if the servos are mounted with their shafts downward. Servo 1 would be suspended from a support anchored to the left or lower left of the origin. Servo 2 would be suspended from servo 1's arm. Miniature casters or some other form of support should be provided at servo 2 and near the pen.

If for some reason it appears desirable to use two arms of different lengths, the angles can still be found in closed form, although the calculation is slightly more complicated.

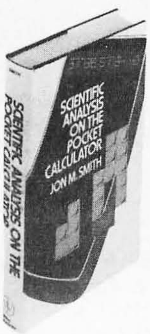
Keep up the good work.

Ronald M Martin, President
Martin Associates
POB M1124
Ann Arbor MI 48106 ■

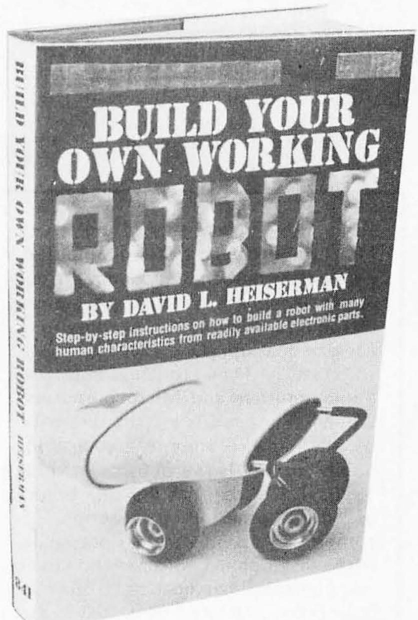
BITS™



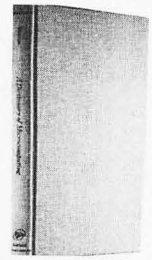
___Game Playing With Computers by Donald D Spencer, published by Hayden. What does it mean to play games using a computer? Read this book to get an introduction into numerous recreational uses of the computer to program and play mathematical and logical games. Topics include numerous mathematical problems, casino games, board games, unusual gambling games, and miscellaneous logic games. Numerous BASIC language programs and listings are included to show details. $16.95.



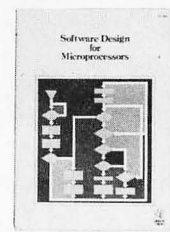
___**Scientific Analysis on the Pocket Calculator** by Jon M Smith, published by John Wiley & Sons. This book is another in a set of source books for mathematical analysis using the contemporary products of technology. It is oriented to the pocket calculator, yet it will provide you with algorithms and methods useful with any personal computer which implements the scientific and analytical functions found on a good pocket calculator. For a more complete description, see the book review on page 120 of the December 1976 BYTE; or order its 392 pages of detailed technical information and review its use for yourself. $13.75.



___**Build Your Own Working Robot** by David L Heiserman, published by Tab Books. This book will not tell you how to build Robbie, the robot of **Forbidden Planet**, or a classical android of science fiction. What it will introduce you to is the problems of making a robot mobile device called Buster III, using pre-microprocessor TTL integrated circuits for all logic functions. It is a must book for background reading, but much of the logic can be extremely simplified using today's microprocessor technology. Use this book as a first look at these problems from which you can build further and more elaborate solutions. Softbound, $5.95.



___**A Dictionary of Microcomputing** by Philip E Burton. In the opinion of BYTE's editor, Carl Helmers, "This is one of the best *designed* and executed dictionaries of computer related terms yet seen on the market. It is of particular relevance to those individuals who want a good general reference to numerous technical terms, broadly covering hardware and software fields as currently practiced." This new hardbound edition is part of the Garland Reference Library of Science and Technology. $12.50.



___**Software Design for Microprocessors.** This stand alone guide to microprocessors has been designed by the people at Texas Instruments to convey knowledge to the first time user of microprocessors. This excellent source book of computer concepts begins with an outline of the basic principles of the general purpose computer, its machine architecture, software, and methods of addressing. It proceeds to discuss how to build software, what is involved in documenting what you've done once you've done it, the mechanics of programming, and specific examples using the TI TMS-1000, TMS-8080, TMS-9900 and SBP0400 designs. You'll find a thick hardcover textbook filled with over 370 pages of useful information including a comprehensive glossary of microprocessor terminology, among several other detailed appendices. $12.95.

Send to:

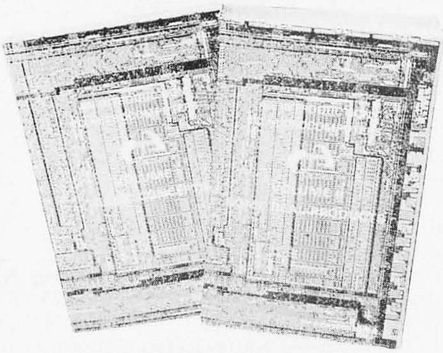
BITS, Inc
70 Main St
Peterborough NH 03458

Total for all books checked $ __________
Postage, 25 cents per book for __________ books $ __________
Grand Total $ __________

Check payment method:
___My check is enclosed
___Bill my MC No. ______________________ Exp. date ________
___Bill my BAC No. ______________________ Exp. date ________

Name ______________________
Address ______________________
City ______________ State ______ Zip ______
Signature ______________________

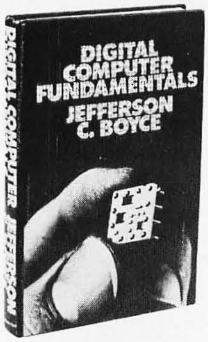
You may photocopy this page if you wish to leave your BYTE intact. Please allow six weeks for delivery.



Adam Osborne's books **An Introduction to Microcomputers, Volumes 1 and 2,** are a concise compendium of the technical details of microprocessors at the component (engineering) level. These are the source books for the system designer who plans to employ the microprocessor, or the advanced homebrewer who wants a dash of customization not found in commercial products.

_____ Volume 1 is subtitled "Basic Concepts." This is the book which presents a framework of ideas concerning the design and use of small computers implemented with LSI. Topics include definitions of the microcomputer, fundamental concepts of logic and numbering characteristics of instruction sets, etc. $7.50.

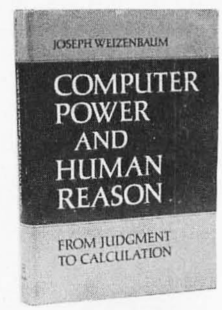
_____ Volume 2 is a much thicker (895 page) detailed volume which complements the information in the first volume. This is the volume which fills in many of the details left out of the conceptual treatment in Volume 1. Here you'll find 19 detailed chapters on the engineering and logical specifications of products made by 16 different manufacturers, including in many cases reprints from the manufacturers' documentation as well as new materials provided by the author. Published in 1976, it even includes such processors as the MicroNOVA by Data General and the Texas Instruments TMS-9900 as well as the older 8 and 16 bit machines. Organization is by design type, and where parts of several manufacturers were intended for a given processor design such as the 8080, these are grouped into a single chapter. $12.50



_____ **Digital Computer Fundamentals** by Jefferson C Boyce. The way to a world of learning is through books. A great place to start, and to return from time to time, is the classical textbook. This new book from Prentice Hall is intended as just that. Topics covered include digital computer operation, basic computer circuits and concepts, Boolean algebra, implementing computer operations in hardware, communicating with the computer and related issues of coding schemes, detailed discussions of the control section, memory functions, arithmetic and logic functions, input and output functions of a classical computer, a chapter on computer programming and a final summary chapter on the details of a typical minicomputer design interpreted in the light of the more theoretical general concepts in the book. This book is excellent background information for the literate and well read hacker. Order yours today. $15.95 hardbound.

_____ **How to Buy and Use Minicomputers and Microcomputers** by William Barden. People have often asked us where to turn to get an introductory book about computers for personal use. One excellent place to start is **How to Buy and Use Minicomputers and Microcomputers,** William Barden Jr's instant summary of the small computer revolution, published by Howard Sams in mid-1976. This is one of the first books of the "general introduction to computers" genre to be published with an emphasis towards the small computer and personal computing as it is being practiced these days. the book, written for the novice as well as the expert, surveys the technical details of the field in nine chapters and 10 appendices. This book is light (but essential) reading for the experienced computer person, and worthy of serious, concentrated perusal by the novice. $9.95.





_____ **Computer Power And Human Reason** by Joseph Weizenbaum. This book is one which should be purchased or read for several reasons. If you're presently a programmer by trade or skill, you'll see a philosophy of computer use and abuse propounded. It's genuinely interesting, and definitely provocative if you reference the storm of letters, counter letters and counter counter letters which this book produced in the Association for Computing Machinery's *SIGART* newsletters during 1976. If you're a novice to the field, the tutorial and explanatory chapters of this book, which are aimed at the layman, will serve as an excellent background source which is also eminently readable. This includes an excellent and low level explanation of what an algorithm is, and how computers go about executing effective algorithms. $5.95 softbound.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

Total for all books checked $ __________
Postage, 25 cents per book for __________ books $ __________
Grand Total $ __________

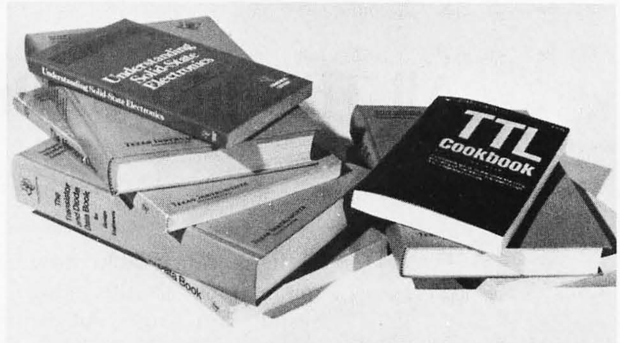
Check payment method:
___ My check is enclosed
___ Bill my MC No. ____________________ Exp. date ________
___ Bill my BAC No. ____________________ Exp. date ________

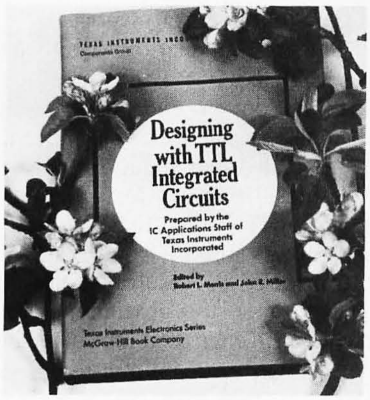
Name ______________________________
Address ______________________________
City ____________ State ______ Zip ______
Signature ______________________________

You may photocopy this page if you wish to leave your BYTE intact. Please allow six weeks for delivery.

A BIT More

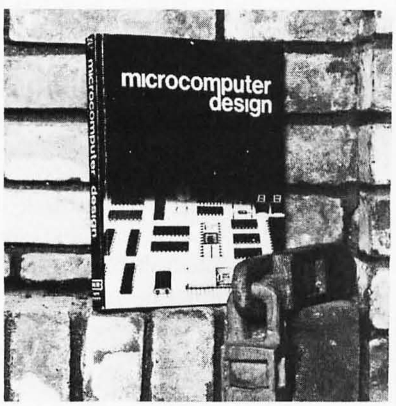
When you build a project, you need information. All you find in the advertisements for parts are mysterious numbers identifying the little beasties . . . hardly the sort of information which can be used to design a custom logic circuit. You can find out about many of the numbers by using the information found in these books. No laboratory bench is complete without an accompanying library shelf filled with references.





Order these absolutely essential references from Texas Instruments today:

- ___The TTL Data Book for Design Engineers, $4.95, new second edition.
- ___The Linear and Interface Circuits Data Book for Design Engineers, $3.95.
- ___The Semiconductor Memory Data Book for Design Engineers, $2.95.
- ___The Transistor and Diode Data Book for Design Engineers, $4.95.
- ___The Power Semiconductor Handbook for Design Engineers, $3.95.
- ___Understanding Solid State Electronics, $2.95.
- ___The Optoelectronics Data Book for Design Engineers, $2.95.
- ___Designing with TTL Integrated Circuits, edited by Robert L Morris and John R Miller, published by McGraw-Hill, $26.



___**The TTL Cookbook** by Don Lancaster, published by Howard W Sams, Indianapolis. Start your quest for data here with Don's tutorial explanations of what makes a TTL logic design tick. 335 pages, $8.95.

___**Microcomputer Design** by Donald P Martin. Edited and published by Kerry S Berland, Martin Research. Purchase your copy of the definitive source for circuitry and hardware design information on the 8008 and 8080 computers today. Even Intel, the originator of the microprocessor revolution, is hard put to compete with the wealth of information found in Martin Research's new second edition of **Microcomputer Design.** This is the book which was originally published as an expensive (but quite practical) engineering report in loose leaf form, at about the time the microprocessor technology was first catching on in the form of the 8008. This 388 page second edition of the manual is loaded with detailed information on how to build and use computers based on the 8008 and 8080. $25.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

master charge | BANKAMERICARD

Check payment method:

___My check is enclosed
___Bill my MC No. ______________ Exp. date ______
___Bill my BAC No. ______________ Exp. date ______

Name ______________________

Address ______________________

City ______________ State ______ Zip ______

Total for all books checked $ ______
Postage, 25 cents per book for ______ books $ ______
Grand Total $ ______

Signature ______________________

You may photocopy this page if you wish to leave your BYTE intact. Please allow six weeks for delivery.

All This Just to Print a Quotation Mark?

David Chapman
Western Reserve Academy
Hudson OH 44236

It is remarkable how much trouble a minor fault in the design of a computer system can cause. An example that I have run up against is the virtual impossibility of printing a quotation mark in several versions of BASIC with any degree of efficiency. The version of BASIC I presently use is Digital Equipment Corp's EDUSYSTEM 25 BASIC, but I've found the same nit in several other BASIC interpreters.

Suppose it is necessary that the following string be output:

IN BASIC, THE QUOTATION MARK (") IS A DELIMITER.

On a cursory examination, it may appear that all that is necessary is:

```
10 PRINT "IN BASIC, THE QUOTATION MARK (") IS A DELIMITER."
```

There is a problem here, however. The second quote, the one in parentheses, will be interpreted as indicating the end of the string. The interpreter is then left with the unexecutable:

) IS A DELIMITER."

and types out an error message.

There are a number of ways to get around this problem. One is to substitute single quotes for double quotes in output strings and hope that the user will understand that they are really double quotes. This is fine for some things, but not for others, like good guy systems in which it is assumed that the user knows absolutely nothing and that substitution would result in confusion.

In this case the programmer finds him/herself confronted with the problem sometimes known as delimiting the delimiter. In BASIC, as you may have gathered, the quotation mark is a delimiter: the delimiter for a literal string; which is to say that it indicates where the string begins and ends. Another example of a BASIC delimiter is the comma, which is used to separate the arguments of functions and of INPUT, READ, DATA and so forth.

The problem is separating delimiters from things delimited which look like delimiters but aren't. One possible solution, unfortunately not available in any BASIC implementation that I have seen, is to use two delimiters for the same function. For example, in SNOBOL, the two literal string delimiters are the single quote and the quotation mark. Hence, using a similar ploy, our example could be coded as:

'IN BASIC, THE QUOTATION MARK (") IS A DELIMITER.'

The other possible solution is to use a roundabout method, printing the quote outside the string. One way of doing this is to use the BASIC CHR$ function. The CHR$ function reduces its argument to a binary number and then dumps it on the output device. Since the 7 bit ASCII for a quote is 0100010, or decimal 34, we can say:

```
10 PRINT "IN BASIC, THE QUOTATION MARK ("; CHR$(34); ") IS A DELIMITER."
```

GLOSSARY

Argument:	The argument or arguments of a statement or function is the value or values which it acts upon. For example, in the expression SIN (2*X), the argument of the SIN function is 2*X.
Delimiter:	A delimiter is something, usually a special character, which tells the language program where something begins or ends.
*** Idiot proof:**	An idiot proof system is one which not even a genius can mess up.
*** Good guy system:**	A system that is both idiot proof and easy to use and learn about.
Literal string:	A chunk of text which is to be used just as it appears in the program.

*Very important terms . . .

```
10 REMARK THIS IS THE PROGRAM WHICH INITIALIZES THE
20 REMARK FILE. IT ONLY NEEDS TO BE RUN ONCE.
30 OPEN "FILE" FOR OUTPUT 'THIS DESIGNATES THE OUTPUT FILE.
40 PRINT#, CHR$(34) 'PUT A QUOTE ON THE FILE.
50 CLOSE 'CLOSE THE FILE.
```

Listing 1a.

```
10 REMARK THIS IS THE MAIN PROGRAM, WHICH PRINTS THE QUOTE.
20 OPEN "FILE" FOR INPUT
30 LINPUT#,A$ 'THIS SETS A$(1) EQUAL TO A QUOTE.
40 A$=A$(1) ' "A$" IS SHORTER THAN "A$(1)".
50 PRINT "IN BASIC, THE QUOTATION MARK ("A$") IS A DELIMITER."
60 REMARK THAT IN LINE 50 THE SPACES AND SEMICOLONS ARE
70 REMARK OMITTED, ALLOWABLE IN THIS BASIC BECAUSE THEY
80 REMARK ARE REDUNDANT DELIMITERS!
```

Listing 1b.

Listing 1: This is the final method needed to output a quotation mark using the DEC EDUSYSTEM 25 BASIC. The program of 1a sets up a file and puts the quotation mark in the file. The listing of 1b uses the file to actually output the quotation mark. In this variant of BASIC, a single apostrophe begins a comment field on a line, avoiding the need of the word "REMARK" in many cases!

This works, but burns time, lineprinter paper, core space and finger tips if used often. The obvious thing to do would seem to be to assign a string variable the value of a quotation mark, so that such a line as this might be used:

> 20 PRINT "IN BASIC, THE QUOTATION MARK ("; A$;") IS A DELIMITER."

but this causes the same old problem if one uses:

> 10 A$=" " "

and furthermore, most BASICs will not allow CHR$ except in PRINT statements.

One way to get around this BASIC limitation is to ask the user to type a quotation mark and input it as a string variable at the beginning of the program:

> 10 PRINT "PLEASE TYPE A QUOTATION MARK ("; CHR$(34); ")";
> 20 INPUT A$
> 30 PRINT "IN BASIC, THE QUOTATION MARK ("; A$;") IS A DELIMITER."

This, however, is not idiot proof. To remove, or at least isolate, the idiot from the system, we can use a separate program, run only once, which puts a CHR$(34) onto a mass storage file. The main program can then open the file later and get the quote back.

This is very devious, but worse is yet to come. In the BASIC that I use, EDUSYSTEM 25, it is apparently impossible to input a quote, from either a file or a keyboard! For some reason known only to the savants at DEC it is possible to use another statement, called LINPUT, to define the quote. In EDUSYSTEM BASIC, strings are limited to six characters in length. LINPUT, short for line input, sets up an array in which each element but the last is a string of length six, and in which the last element is a string which contains whatever is left over.

Listing 1 is a final solution in EDUSYSTEM 25 BASIC. ■

The Moral

All this, just to print a quotation mark! Similar, nitty details are often found in any large piece of software — it's in the nature of the beasts. Use of a program often requires adaptability and ingenuity!

Goes National

during the 1977 NCC, June 13–16

Personal Computing Fair & Exposition

The fast-growing field of Personal Computing will share the national spotlight in June, when the 77NCC will recognize the dynamic growth and promise of the field with the Personal Computing Fair & Exposition. In addition to the commercial exhibits of Personal Computing manufacturers, dynamic displays and demonstrations of non-commercial individual and group-owned projects will be featured at the Dallas Convention Center. The success of other hobbyists can give you new ideas for your own systems, "how-to" tips and dozens of clever solutions to everyone's problems. You might even find a joint-venture partner with a kindred spirit. More than 100 non-commercial small computing systems are expected, featuring hardware and/or software implementations, games, recreation, music, art, amateur radio, scientific, miscellaneous and general applications. Prizes and awards will be given in all categories.

Personal Computing Program

Two full days of panel sessions on June 15 and 16 will provide an in-depth look at Personal Computing: Past, Present and Future; The Future of Retail Computer Stores; Hardware of the Computer "Hobby" Market; and Personal Computing Software. Leaders in the Personal Computing movement will appear on each of the panels to let you know the latest developments in the field, point out trends you'll need to be aware of—and answer your questions.

Special Interest Sessions

In addition to the panel sessions, special interest groups will be able to gather informally for "how-to" programs on building a kit, debugging software, using assembly language, I/O interfaces, cassettes and disks, software standards and so on, into the night. If the special interest group you want is not organized when you get there, we'll do our best to help you get one started!

National Club Congress

Is a national personal computing association needed? If it is, what does it do, how does it do it, and who does it? To find out what's happening—pro and con—club reps from across the nation will gather to exchange ideas and discuss issues related to club activities and programs. Make certain your club sends an official delegate who can speak for you and vote vis-a-vis a national organization, establishment of national hardware/software standards, a national program library and interchange, educational seminars, meetings, *ad infinitum.*

1977 NATIONAL COMPUTER CONFERENCE
Dallas Convention Center • June 13-16

8080 Programming Notes

Paul Krystosek
57 N Lincoln
Lombard IL 60148

John McCarty
5412 N Arrow Dr
Peoria IL 61614

With a couple of exceptions the 8080 microprocessor instruction set primarily operates on eight bits at a time. But with some imagination several 16 bit operations can be performed by using these exceptions. The basic technique is to use the HL register pair as an accumulator rather than as an address and index register. The simplest 16 bit extension is just a different name for the DAD H instruction. The effect of this operation can be stated in three ways:

A Add the HL register pair to itself.
B Multiply the HL register pair by two. (We might rename it M2HL.)
C Shift the HL register pair left one bit position. (We might rename it ASLHL for "arithmetic shift left HL.")

These three statements are logically equivalent. To multiply by greater powers of two just repeat the operation the required number of times.

So far the DAD H instruction has been referred to as an arithmetic shift rather than a rotate. The carry bit is set as in a rotate, but it does not affect the low order bit. To set up a 16 bit rotate, increment the HL register pair after the shift if the carry bit is set, as shown in listing 1.

line number	label	mnemonic	commentary
1		DAD H	HL:=HL+HL;
2		JNC DONE	go to DONE if carry is zero;
3		INX H	HL:=HL+1;
4	DONE	EQU *	first line of routine DONE;

Listing 1: Performing a 16 bit rotate left using the HL register pair. The DAD H instruction will shift the 16 bit number 1 bit to the left and determine the value for the overflow flag. If the overflow flag is set, then the HL register pair is incremented by 1.

16 bit addition is of course done by using the DAD B or DAD D instructions which exist in the 8080. In such cases, one number to be added is first loaded into the HL register pair, and then the other number is loaded into the BC or DE register pair and added to the HL register pair. A common use for this instruction is in index address calculation. A special use for this instruction is in looping and comparing. When setting up a loop using a counter which will have a value greater than decimal 256, a multiple precision increment operation using two single independent registers can be used. Note that since the INX instruction does not affect any status bits, it cannot be used by itself as a 16 bit loop control function.

line number	label	mnemonic	commentary
1		LXI B,0	BC:=0;
2	LOOP	EQU *	
		.	[main body of loop]
		.	
		.	
n		INX B	BC:=BC+1;
n+1		LXI H,–1000	loop to stop at 1000;
n+2		DAD B	HL:=HL+BC;
n+3		JNC LOOP	go to LOOP if carry is zero;

Listing 2: An example of setting up a 16 bit counter using the BC register pair. In this example, a loop will go through 1000 iterations.

Another approach is shown in listing 2. The –1000 on line n+1 will be stored as a two's complement number hexadecimal FC18. When adding anything less than decimal 1000 in statement n+2 a carry will not result. Statements n+1, n+2, and n+3 could be thought of as "compare the HL register pair to the BC register pair and jump if the

BC register pair is less than the HL register pair."

If a compare high, low or equal is needed, the procedure could be extended as in listing 3. A disadvantage of this technique is that it requires the HL register pair to be loaded with the two's complement of the target number. It is probably easier and certainly takes less execution time if we load the two's complement of a constant during assembly instead of a variable calculated during execution.

Another modification of the loop routine, shown in listing 4, can be used when processing or scanning consecutive memory locations where the end of the loop is determined by an address rather than a counter. An example of this is in outputting to a port from one address to another, as would be done when writing a block of data to tape, disk, another machine, or whatever.

The (−ADR2) on line 7 may look confusing; it is just the two's complement of the address in question. While it would be slow to hand assemble such a statement, a good assembler will do the calculation for you. To have variable start and end addresses the two's complement would have to be calculated by subtracting ADR2 from zero, or by a binary complement operation on the 16

```
label    mnemonic       commentary

         LXI H,-1000    HL:=(-1000);
         DAD B          HL:=HL+BC;
         JNC LOW        go to LOW if BC register pair less than 1000;
         MOV   A,H  )
         ORA L      >   test if HL register pair equals zero;
         JZ EQUAL   )
HIGH     EQU *          beginning of routine if BC register pair greater
                          than 1000;
            .
            .
            .
EQUAL    EQU *          beginning of routine if BC register pair equals 1000;
            .
            .
            .
LOW      EQU *          beginning of routine if BC register pair less
                          than 1000;
```

Listing 3: This routine is an extension of the testing used by the routine of listing 2. Here, the goal is to pick one of three branch possibilities: less than, equal to, or greater than. This routine is used to test register pair BC's relationship to the integer constant 1000 using DAD H with HL as a 16 bit accumulator.

```
line
number  label    mnemonic       commentary

 1               LXI B,ADR1     BC:=ADR1;
 2      LOOP     IN CTL      )
 3               ANI STATUS  >  loop waiting for data ready for output;
 4               JNZ LOOP    )
 5               LDAX B         A:=BC;
 6               OUT DATA       output data;
 7               LXI H,-ADR2    HL:=(-ADR2);
 8               DAD B          HL:=HL+BC;
 9               INX B          BC:=BC+1;
10               JNC LOOP       go to LOOP if address in BC register pair less
                                  than ADR2;
                    .
                    .
                    .
 n      ADR1     EQU *          starting address;
 n+1             DS             define storage area;
 n+2    ADR2     EQU *          ending address;
 n+3    CTL      EQU 2          equate CTL to 2;
 n+4    DATA     EQU CTL+1      equate DATA to CTL+1;
 n+5    STATUS   EQU 80H        equate STATUS to bit pattern to be compared
                                  from input port;
```

Listing 4: This routine is basically the same as listing 2 in determining when to exit the loop. The main difference is that an address is being used instead of an incremental or decremental counter. This particular routine outputs data from address ADR1 to address ADR2.

```
label    mnemonic       commentary

         LXI B,-1       BC:=(-1);
         LXI H,1000     HL:=1000;
LOOP     DAD B          HL:=HL+BC;
         JNC DONE       go to DONE if carry equals zero;
            .
            .
            .
         JMP LOOP       go to LOOP;
            .
            .
            .
DONE     EQU *          start of routine where the loop exits to when done;
```

Listing 5: Here is a loop with a 16 bit counter which, instead of adding and comparing two numbers to find when to exit, decrements the loop counter and checks to see when it goes past zero to negative one.

A Note About "EQU"

Throughout the listings of this article the mnemonic EQU has been used extensively. The EQU, equate, statement is an assembler instruction or "pseudooperation" rather than a machine instruction. Pseudooperations are common assembler features which tell the assembler something about your program or the assembly process but do not necessarily generate machine code. The asterisk (*) or dollar sign ($) (depending on the assembler that is used) indicates "this address." For example:

```
1) LOOP EQU *
        .
        .
        .
n) JMP LOOP
```

The effect of statement 1 is to equate the address of the next instruction with the label LOOP. The value of the address is then stored in the assembler's symbol table. This may seem like a trivial thing to do; but if you wanted to use your text editor program to add an instruction at the beginning of the loop as a program patch it can just be put there. Whereas if the label was on the first instruction that line would have to be changed as well as the new line being inserted. *[Of course if your text editor is capable of arbitrary insertions this is not a major point.]*

Another use of the equate pseudooperation is the following:

```
1) CTL EQU 2
2) DATA EQU CTL+1
```

The effect of this would be that whenever CTL is encountered it is replaced by the value 2. Statement 2 would equate DATA to 3 wherever it is used. The real usefulness of equates is in their ease of change at the assembly language level. If the IO port address is changed, only the equate for CTL would have to be changed to make the change throughout the program for both CTL and DATA the next time it is assembled. Table lengths can be handled with a similar ease as shown in the listing below. In this way, if the length of the table is changed, the change will be made throughout the program the next time it is assembled, providing of course that TLEN is used consistently.

Another handy fact is that if you have used two different spellings for the same label or variable, one can be equated to the other rather than changing them all to one.

line number	label	mnemonic	commentary
1	TBL	DW 'BA'	define character strings; [misc data for demonstration]
2		DW 'CD'	
3		DW 'FE'	
4		DW 'HG'	
5	TLEN	EQU *–TBL	equates TLEN to the current address – beginning address;

bits followed by a 16 bit increment operation.

Yet another application of the double add instruction with two's complement arithmetic is in decrementing a register pair with an indication of passing zero. The two's complement of the number –1 is all 1 bits regardless of how many bits are in the number. The only number which can be added to –1 without causing a carry is zero. Adding –1 to a number will decrement that number by one and set the carry flag unless the number was zero. Using these facts a loop can be set up as in listing 5.

When using this technique, the decrement and check if zero are done at the beginning of the loop as in languages like PL/I rather than at the end of the loop as in FORTRAN. The reason for this is that we find out when the loop goes from zero to –1 rather than when it is zero. A disadvantage to this technique is that it ties up a register pair to hold the –1 value.

The techniques illustrated here are just a few examples of what one can come up with by using a little imagination and looking very closely at exactly what each instruction does. They are not the most general techniques as far as usefulness in many situations, but used in the right place they can save a few bytes or microseconds. If nothing else, these notes have shown ways to use instructions that may not be immediately obvious.■

REFERENCE

Intel Corporation: *Intel 8080 Microcomputer Systems Users Manual,* September 1975.

Here are 10 reasons why *you* should subscribe to *CREATIVE COMPUTING* today.

1. **Creative Computing will help you decide which computer is best for you.**

Creative's no-nonsense equipment profiles arm you with the facts before you talk to the vendor or dealer. Whether you're interested in a microcomputer kit, a mini, terminal, or programmable calculator, you'll find the evaluative information you need in *Creative*. Indeed, one wise hardware decision could save you the cost of a lifetime subscription!

2. **Creative Computing discusses computer applications in non-technical, understandable language.**

Business people who want to know what's going on in the EDP department, students who want to learn about microprocessors, hobbyists looking to make good use of home computers, or anyone concerned about the effect of the computer on society will find these and many, many more mind-expanding topics covered on the pages of *Creative*.

3. **Creative Computing covers computer education in depth.**

After all, that's where we got our start and so we continue to present four or five major learning activities every issue. If you're a teacher, *Creative* will save you hours of preparation time. If you're a student, you'll be way ahead of your class with *Creative*. And if you've already graduated, you can bone up on what you missed.

4. **Creative Computing carries outstanding fiction every issue.**

One of the best ways of exploring future scenarios of computer usage is through fiction, so *Creative* seeks out material from the best authors — Isaac Asimov, Frederik Pohl, Arthur C. Clarke to name just a few, as well as many others who are destined to be the best of the next generation.

5. **Creative Computing's book reviews are honest and timely.**

We're not owned by a big book publisher to whom we owe loyalty, nor do we depend upon advertising for our revenue. Hence, not only do our reviews pull no punches, but we also rank order similar books (like all 34 books on the BASIC language which we reviewed last year). *Creative* reviews virtually every computer book of interest to educators, hobbyists, and general laypeople, even including vendor manuals and government pamphlets.

6. **An extensive resource section will save you time and money.**

Every issue of *Creative* carries 40 or more short resource reviews evaluating periodicals, booklets, hardware, terminals, couplers, peripherals, software packages, organizations, dealers, and much more. Every entry has a brief description, evaluation, and the name, address, and phone number of the vendor. You'll save valuable time seeking out this information, much of which you'd possibly never come across.

7. **Creative Computing will provide hours of mind-expanding entertainment, even if you don't have a computer.**

Creative Computing carries 10 or 12 pages of games and puzzles every issue. Most of the puzzles don't need a computer or calculator to solve; some do. Naturally, the 4 or 5 new computer games (in Basic, Fortran, and APL) in every issue require access to a computer.

8. **Creative Computing gives you things to actually do with a computer.**

Home computer kit, mini, timesharing terminal — whatever your access to computer power, *Creative* provides thoroughly documented programs with complete listings and sample runs that you can use with minimum effort. Games, simulations, CAI, computer art — whether education or recreation is your bag, you'll find programs which you can use in *Creative*.

9. **A no-compromise policy of editorial excellence means every issue is of value to you.**

We firmly intend to be around a long time and we believe the way to do that is to publish only material of the very highest quality. We believe our readers are among the most astute, intelligent, and active people in the world. You can't afford to waste time reading imprecise, opinionated, or wordy articles and you won't find any in *Creative*.

10. **The price is right — only $21 for 3 years.**

That same $21 will buy you a pair of Star Trek walkie talkies, six direct dialed 10 minute calls between New York and Boston, 3 tankfulls of gas, or 10 cocktails at a Hilton hotel. Wouldn't you rather have 18 issues of Creative Computing each containing over 85 pages of solid editorial material (including advertising, over 100 pages per issue). Count the editorial pages in *any other* hobbyist or special interest magazine and compare it to *Creative*. *Any other* 1 year subscription $8. Lifetime $300.

NO RISK GUARANTEE

You may cancel your subscription at any time for any reason and we will refund the balance without question.

David H. Ahl, Publisher

FOR FASTER RESPONSE

You can phone your subscription orders to us at (201) 366-5617. Our lines are open from 9 am to 4:30 pm EST.

SUBSCRIPTION ORDER FORM

Type	Term	USA	Foreign
Individual	1-Year	□ $ 8	□ $ 10
	3-Year	□ 21	□ 27
	Lifetime	□ 300	□ 400
Institutional	1-Year	□ 15	□ 15
	3-Year	□ 40	□ 40

□ New □ Renewal

□ Cash, check, or M.O. enclosed

□ BankAmericard Card No. ______________

□ Master Charge Expiration date ______________

□ Please bill me ($1.00 billing fee will be added)

Name ______________

Address ______________

City ______________ State ______ Zip ______

Send to: Creative Computing, Department B
P.O. Box 789-M, Morristown, NJ 07960

Technical Forum

Improving Quadratic Rehash

"Making Hash With Tables" by Terry Dollhoff *[BYTE, January 1977, page 18]* is a good introduction to hash tables. However, quadratic methods for collision avoidance do not have to be complicated or suffer from "half table search." If the table length is a power of 2 and the quadratic increment is 3 as in the following simple and fast algorithm, then none of the table will be excluded from the search.

I have been using this scheme since about 1970 but have never seen it reported in the literature. Your readers may write to me for a copy of the proof that it works.■

John F Herbster
Herbster Scientific
3233 Mangum Rd, 100
Houston TX 77092

A Quadratic Hash Table

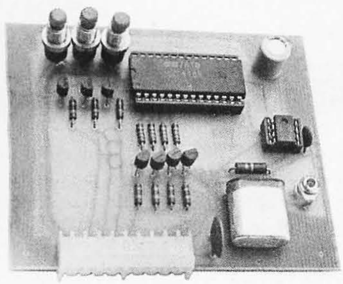
The following algorithm assumes that the table length is a power of 2, the table words were initialized to VIRGIN, and MASK has a value equal to the table length minus 1.

1. Set DEL to 0.
2. Set I to hash code of KEY.
3. LET I=I.AND.MASK (eg: AND I with MASK).
4. IF TABLE (I)=VIRGIN then go to NOTFOUND. (Note that TABLE(I) refers to the contents of location TABLE+I).
5. If TABLE(I)=KEY then go to FOUND.
6. Let DEL=(DEL+3).AND.MASK.
7. If DEL=0 then go to FULL. (Note that DEL gets back to 0 only after the whole table has been searched.)
8. Let (I=I+DEL).AND.MASK.
9. Go to step 4.

On return to the user's program via the NOTFOUND, or FOUND exits, the index, I, will point to the spot for a new table entry or the found entry respectively. The FULL return means the KEY was not found and that the table is full. Note that the value VIRGIN may not equal any possible value of KEY.

What's New?

Keep Track of Real Time Now



(What Other Kind of Time is There?)

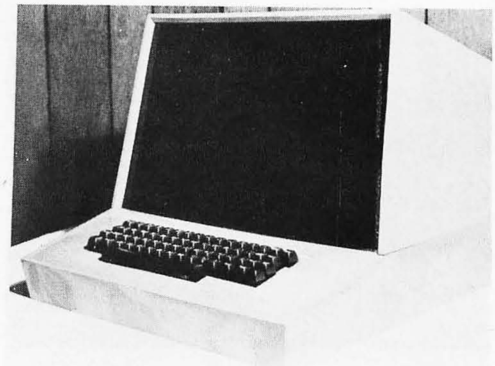
TED, POB 4122, Madison WI 53711, has announced this little real time clock accessory for your personal computer, a small board with a National Semiconductor MM5318N clock chip, interface drivers and a trimmable crystal oscillator. The board requires 4 parallel input lines and 3 parallel output lines (TTL compatible) and power supplies of +5 V and +12 V. Using pushbuttons, you can set the time and forget about it, especially if you use a battery backup to keep the oscillator and clock chip running through power failures.

This clock is sold assembled and tested, and comes with a manual and interface connector. The cost is $39.95 plus $.75 postage. When you hook it up to a single 8 bit PIA parallel port (or equivalent) you'll be able to keep track of the time of day for household automation, use it as a timer in game applications, or even implement such exotic software as a personal events calendar with automatic reminders.■

Cambion Patch Cords . . . and Things

Cambion, 445 Concord Av, Cambridge MA 02138, has sent us a flier on what will probably be a most useful test and prototyping gadget for those who use wirewrap sockets. The product is a set of different length patch cords which have female connectors at each end, for sliding over wirewrap posts which are 0.025 inch (0.064 cm) square. These are ideal for experimental last minute patches in complicated wirewrap panels, and will also prove useful for permanent jumpered options which can be quickly changed. They can only be used for single link jumpers, since no provision has been made for multiple terminal buses. They are part of the Cambion XQ line of packaged parts for retail sale, a line which computer stores should find a good drawing card. The list price for a typical package of five jumpers with 2, 4 or 6 inch (5, 10 or 15 cm) lengths is $8.15. Other lengths and assortment packages are also available. For more information on this and other products in the XQ line, write to Cambion and ask for XQ products pamphlet and the latest price list. The XQ line ranges from precut and stripped wirewrap wire ($5.58 per 100 3 inch [7.5 cm] lengths) to manual wirewrap tool ($3.50) to electronic terminals, to an experimenter's thermoelectric module ($19.95) which acts as a solid state heat pump which can be used to cool everything from beverages or transistors depending on how you apply it.■

A New Desk Top Video Computer



Robert Boyer, who is starting a firm called Computer Data Systems of Newark DE, sends this picture of a new product they plan to market in the near future. The specifications according to the press release accompanying this picture include:

- 9 inch high resolution monitor
- 63 key ASCII encoded keyboard
- video display board providing 16 lines of 32 characters
- 10 A power supply
- 8080 processor with ROM operating system, real time clock, cassette interface
- Altair compatible bus with 10 sockets
- Commercial CRT cabinet with smoke grey screen and cooling fan.

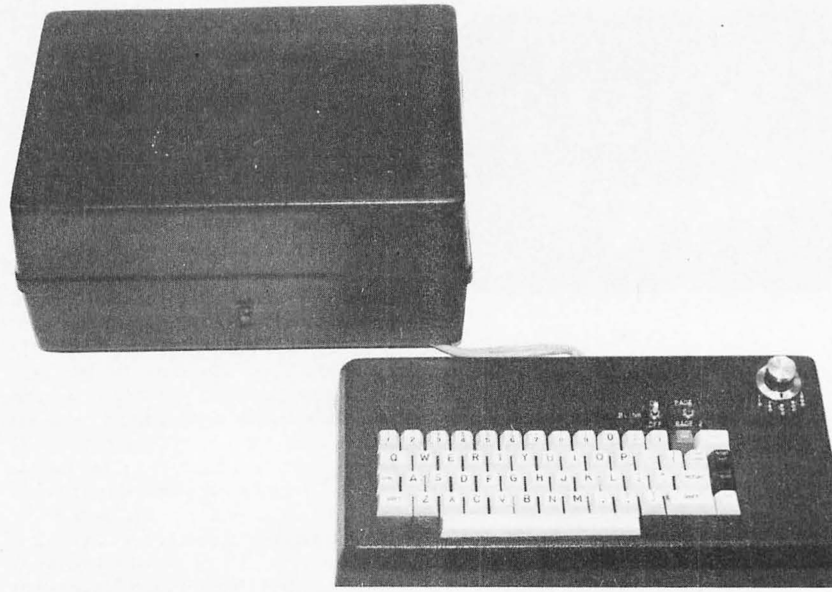
The press release describes this prod-

uct as "The Versatile CRT," which will be available in kit form at $995.95 plus shipping. Availability is projected for early 1977. For more information, contact Robert Boyer at Computer Data Systems, English Village Apts, Atram Hall #3, Newark DE 19711.■

Why Is This Fellow Smiling?



Because he's Pete Roberts, owner of Computer Kits Inc, the MITS distributorship in Berkeley CA, and he's celebrating the start of his second year in business according to the press release accompanying this picture. He's doing well enough to be contemplating the opening of three additional stores in the San Francisco Bay Area to complement his present store in Berkeley at 1044 University Av.■



A Southwest Technical Products Coverup, Revealed . . .

Parsec Electronics is now marketing an enclosure kit for the SWTPC CT-1024 Terminal System. A two enclosure design is used to give the terminal suitable protection while retaining the flexibility of a movable keyboard for the user. The two cases are formed from strong, resilient ABS plastic and are "color coordinated" to blend in with the SWTPC 6800 computer system. The keyboard enclosure has the cutout and mounting pads for the KBD-5 keyboard and a fully enclosed bottom. The main case features mounting pads for the CT-1024 main board, the GT-61 graphics board, the AC-30 cassette interface board and room for their power supplies. The kit includes mounting hardware for all circuit boards, a heatsink for the power supplies and rubber feet for both enclosures. For more information contact Parsec at POB A82327, San Diego CA 92138.■

BYTE'S FOREIGN SUBSCRIPTIONS

BYTE has more foreign readers than ever before. The magazine has almost doubled in weight, and is expected to increase further. Also, postal rates have gone up. In view of these increases we need to raise our foreign subscription rates effective May 1, 1977:

CANADA and MEXICO: For one, two, or three year subscriptions, add $5.50 per year to the US subscription rates.

WESTERN EUROPE: For a one year subscription only, add $13 to the US subscription rate (total $25). The magazines will be shipped by air. Present subscribers in western Europe who wish to convert their subscription to air delivery, please contact the publisher.

OTHER FOREIGN COUNTRIES: For a one year subscription only, add $13 to the US subscription rate (total $25). The magazines will be shipped by surface mail from selected distribution centers in the US or abroad.

For additional air rates, please inquire.

We hope that our new distribution routes will expedite the delivery of BYTE. Please let us know of any excessive delays in the delivery of your issue. Thank you.

BYTE Subscriptions
PO Box 361
Arlington MA 02174
U S A (617) 646-4329

Please remit subscriptions in U.S. funds. Thank you.

Solution and Comments to "An 8080 Bug in the Stack," April 1977 BYTE, page 161:

A programmer's reaction is to first check out SVREG and GTREG to see if the stack is being used improperly or to see if the PUSHs and POPs somehow disturb the order of saving and restoring registers. This is a good exercise in the use of the 8080 stack instructions, but reveals nothing.

Almost instinctively, the next trial fix is to change the code to

```
SUBRX:  CALL    SVREG  ;SAVE ALL
          .
          .
          .
          .
        CALL    GTREG  ;RESTORE ALL AND RETURN
        RET
        ;
```

Now, the two subroutines perform correctly, but it is not clear why until the programmer really digs into the stack manipulations. When enough digging is done, it finally turns out that GTREG expects a return address at the top of the stack, but there is none if JMP GTREG is used. Instead, the top of the stack contains the saved contents of double register D,E and the desired return address is actually buried down in the stack and will not be available until the stack pointer is incremented 8 times (by 4 POPs).

One could also fix GTREG so that it is *always* entered with a JMP (and never a CALL), but the way it was originally coded, GTREG must always be entered with a CALL.■

Blabberwacky: Answer to Bar Code Puzzle

Here is the answer to the bar code puzzle posed on page 130 of the March 1977 issue of BYTE. We'll have a report on the contest winner prepared in time for the July issue, after the deadline of March 31 has passed. As this note is written, we have received word of two promising designs for readers, one prepared by an associate of Chuck Adams at Texas A&M University, and another prepared by Fred Merkowitz, who wrote an article on the subject of signal processing circuits in the December 1976 BYTE *[page 77]*.

'Twas brilliant and the slimy toads
Did go and gander in the wave:
All misty were the bluish groves.
And the mole runs outraged.

"Beware the Blabberwack, my son!
The words that bite, the barbs that catch!
Beware the mynah bird, and shun
The frivolous Bendersnitch!"

He took his vicious words in hand:
Long time the slanderous foe he sought
So remained he by the tilting tree,
And stood awhile in thought.

And, as in malign thought he stood,
The Blabberwack, with eyes of flame,
Came whistling through the stately wood,
And bluffed as it came!

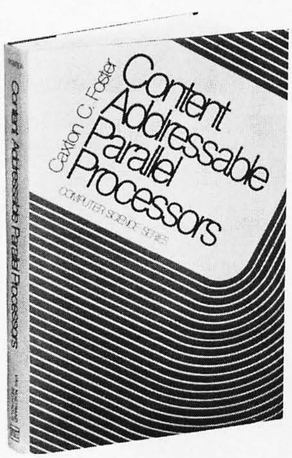
One, two!, One, two! And through and through
The vicious blade went crushing-crash!
He left it dead, and with its head
He went galloping back.

"And hast thou slain the Blabberwack?
Come to my arms, my bouncing boy!
O fabulous day! Hoorah! Hooray!"
He chortled in his joy.

'Twas brilliant and the slimy toads
Did go and gander in the wave:
All misty were the bluish groves.
And the mole runs outraged.■

Book Reviews

Editor's Note: *A homebrew content addressable parallel processor design would make an excellent experimental topic for an article or two (or n), and tutorials on the subject would provide excellent materials for BYTE.*



Content Addressable Parallel Processors *by Caxton C Foster, Van Nostrand Reinhold, New York, 1976, 233 pages, $11.95.*

Do you know anybody named Frank?

– Oh, sure, Frank Smith, you know, . . .

– Stop. How did you know? Let's try a few models:

1. Consider everybody you know, stopping when you hit Frank.
2. Perform a binary search on your acquaintances, whom you keep in mind, sorted by first name: perhaps you hit Nancy, then George, then David, Fred, Frank.
3. Mentally compute a hash address on "Frank," plucking him from a linked collision overflow list in an average of 1.37 probes.

If you're a computer, that's how you know, or by some similar method, whether you're looking for names, syntactic tokens, transaction numbers, or whatever; that is, you do unless you are a CAPP (see title).

I had heard of and talked about parallel processing but had always thought before in terms of multiple processors locked into some sophisticated intercommunication scheme requiring finicky timed programming with all the trimmings. The main revelation in Foster is the simplicity with which true content addressability and parallel operation may be obtained. My main trouble, if I can call it that, is that although the concepts and techniques have been around for 20 years (see the preface), CAPPs remain obscure and not in general use.

Take an ordinary word of memory; add a bit to the end; add a couple of gates to each bit of the word to turn on and off the new bit by comparing the present values of the bits to the information on some data bus. Furthermore, connect the new bit back into the write enable circuitry, so it can control the writing into that word. Now throw away the address circuitry; you don't need it any more. Your word can now be addressed by its *contents.*

To find Frank in your table, put him on the data bus in the proper field of the word and activate the compare lines. Bingo! One instruction table search. (CAPPs tend to have long words for obvious reasons.)

Select your favorite algorithm. Replace the branches with instructions that select the proper words in memory, and in the time it takes to perform each step in sequence, you can execute the algorithm on every word of memory simultaneously. That's what the parallel processing part of the name means.

If you are busy laughing at all the holes in this architecture and explanation, read the book; I gave only the most cursory of explanations. The CAPP as hardware has, it seems to me, the same advantages that APL has as a language: the ability to perform in single operations what would otherwise be machine time consuming loops, or programmer time consuming complex algorithms or klugey hacks.

Just to get in some actual criticism, I finished chapter 8, on STARAN (the only commercially available CAPP), still waiting for the other shoe to drop. It seemed to contain all details and no gestalt descriptions of the machine instructions but no sample programs (exception: the add fields instruction's inner loop), nothing about who has the things, what they do with them, etc. This is just carping, however; the book is in general well written and contains material of interest on every level from hardware through analysis of algorithms to expositions on applications. I recommend you buy, beg, borrow or steal the volume and devour it at your earliest opportunity.

J Storrs Hall
16 Easton Av
New Brunswick NJ 08901

32 K

PERSONAL MINICOMPUTER SYSTEMS.

32 K CPU,
Two CRT's,
And Two
Floppy Disk
Drives With
262 K Bytes
Each.

For Details:
ABBOTT COMPUTER CO.
57 Gregory Street
Marblehead, MA 01945
617-631-8903

8080 Interactive DEBUG

NOW AVAILABLE: A real-time program debugger for all 8080 processors. Debug your code in one-tenth the time it takes by hand. This is not a simulator but a real-time programming aid that until now was only available on mini & macro computers. And it FITS IN 5K!

FEATURES:

*Six I/O modes–Mnemonic, single & double byte numeric, ascii, or dis–placed mnemonic & numeric.
*Four user–defined breakpoints.
*Relocatable to anywhere in memory.
*User–controlled number base–binary, octal, hex; any radix from 2 to 36.
*Commands for displaying & modify–ing all registers–including PSW & SP.
*Search command with masking.
*Block memory move & fill.
*On–line calculator/expression eval'.
*Displacement register for examing relocatable code.

DEBUG 8080 To Order:

(1) Specify: CUTS cassette $30
or Paper Tape $40
or Manual Alone† $10
or Descriptive brochure $ 1
(2) Your hardware: Altair, Imsai, Sol,...
(3) Send to: Bay Area Timeshare
P.O.Box 2298
Berkeley, CA 94702

†refundable on purchase of program

NOW HEAR THIS!!

ALPHA DIGITAL SYSTEMS

SELLS IMSAI KITS AND TOTAL INTEGRATED SYSTEMS ASSEMBLED, CHECKED-OUT, AND WARRANTED AT KIT PRICES.

TYPICAL SYSTEM

* IMSAI 8080 Computer	$ 699
* 22 Slot Mother Board	52
* All Connectors	140
* 4 K RAM	139
* MIO	195
* 3 Cable A	54
* 1 Cable M	12
* Assembled System Total	$1291

ALPHA I

SPECIAL DOS SYSTEM DEAL (includes)
* IMSAI 8080 Computer
* 22 Slot Mother Bd. with Conn.
* 16 K RAM (4 4K Boards)
* 90 K Disk (with Controller)
* DECwriter typewriter 30 cps
* DOS-BASIC Software

Assembled System Total $3995.
(Less DECwriter) $2495.

Select any kit or system of kits from the IMSAI price list, order from us and receive the assembled unit for the same price. If you don't have a price list, drop us a line and we will send you one. Terms: Cash with Order - Prices include freight. (N.C. Residents Add 4% Sales Tax)

How can ALPHA DIGITAL SYSTEMS do all this? Its simple. ALPHA DIGITAL WANTS TO BE YOUR COMPUTER COMPANY.

ADS ALPHA DIGITAL SYSTEMS
RT. 4 BOX 171A
BOONE, N.C. 28607
(704) 264-7946

Spring into Season with a BYTE T-shirt



At last! No more wardrobe crises! BYTE T-shirts are here! Now you have the perfect garb for computer club meetings, Altair Conventions, playing Shooting Stars and computer chess. (A pair of trousers from your own closet is suggested as an addition to the BYTE T-shirt. BITS can't do everything for you.)

BYTE T-shirts are of top quality 100% cotton or cotton-polyester. The original design, by artist Judy Lee Rehling, is silk screened in red on white shirts with blue trim on collars and sleeves, or on blue heather shirts.

The $5.50 price includes postage and handling.

Send to: **In unusual cases, processing may exceed 30 days.**

BITS, Inc.
70 Main St.
Peterborough NH 03458

Please send me ___extra large ___blue heather
___large ___white with blue trim and red letters
___medium
___small T-shirts @ $5.50 each (includes postage and handling)

Total enclosed $________
Bill MasterCharge No.________ Exp. Date________
Bill BankAmericard No.________ Exp. Date________
Name________
Address________
City________State________Zip________
Signature________

All orders must be prepaid
Prices shown are subject to change without notice.

Ask BYTE

CAN I CONVERT PROGRAMS?

I am 13 years old and my dad recently purchased a KIM-1 microprocessor. I have seen a lot of BASIC programs and I wish to know if there is any way to convert these programs to assembly language or, say, MOS Technology machine code.

Kevin Schnitzius
308 Stockley St
Rehoboth Beach DE 19971

Yes, there is. It involves sitting down with the program, figuring out what it does, then doing the same things using the machine or assembly language. In many cases, the BASIC version of a program will take up much more memory than its machine language equivalent, but even so for numerous games you'll want to have a fair amount of memory in the system. The conversion could be compiled automatically on a large machine with an appropriate translator program if it were available, but you'll probably have to do it the hard way by hand. ■

SYSTEMS INFORMATION NEEDED

We need help, more specifically, an education! We presently own a PDP-8-E, with 20 K, dual DEC tapes and four Teletypes. We need to find a source of clearly written directions or information that will allow *us* to do the following without hiring a specialist:

1. Choose the right modems which will allow us to move a Teletype around the school and use the school's 2 digit phones (or any phone system).
2. Hook up the modems correctly.
3. Decide whether we can interface an inexpensive floppy disk to the system, how to do it, and what parameters and ramifications we must face.
4. Decide whether we can interface other inexpensive peripherals shown in BYTE to our system.
5. Write handlers for peripherals.

There seems to be too much expense involved if we go through DEC. We have the programming ability and the mechanical ability, but it ends there.

We would appreciate any help you could give us. Thanks for your cooperation.

Richard A Brown
Instructional Computing Director
Minnechaug Regional High School
621 Main St
Wilbraham MA 01095

Would any reader be able to help out Richard in his quest for information? Someone with familiarity with DEC equipment and intercom style telephone systems would be best qualified. ■

LINEAR EQUATIONS

In my business I use a commercial timesharing system to solve a system of 1000 nonlinear simultaneous order 36 difference equations (but with an average of only ten terms in each equation). This requires about 30 seconds in a core memory of only 600 K bytes. Since reading your magazine I have wondered whether this problem could be solved with any of the larger home-sized computers, say 64 K bytes core, some floppy disks and cassettes, allowing as much as 24 hours continuous running for the job. Would you have any idea whether this is possible, assuming all the software were available? (I believe this would require a kind of software not yet marketed for microcomputers, something similar to the virtual memory timesharing system monitors which keep most of a job's data on high-speed disks while feeding portions of it into core and back to disk many times in the course of the run.)

George Lyons
280 Henderson St
Jersey City NJ 07302

Yes, it sounds possible, provided you always treat subsets of the big array at any given time. However, you must note that several large computer operating systems features may have to be explicitly coded in your BASIC application program on the typical small computer. The problem will be quite solvable given a computer with an extended BASIC interpreter and a floppy disk for intermediate storage. Where you used a 600 K region of memory for large arrray storage in the large system case, in the small system case you incorporate explicit calls to the disk operating system to roll data in and out of memory buffers as needed when referencing subsets of the array. The question of how long it will take depends upon your programming strategy, the characteristics of the floating point package in the BASIC interpreter, and the design of the floppy disk interface. (Does it use interrupts and DMA, or does it require dedication of the processor to IO management during IO?)

Authors Take Note: *Techniques of handling such large arrays in BASIC with disk facilities would make an excellent topic for several articles.* ■

To Dean Blondefield:

I read in the January issue of BYTE in the "Ask BYTE" column that you are inter-

ested in books about robots. I just finished reading a book entitled *Build Your Own Working Robot* by David L Heiserman, published by Tab books (Blue Ridge Summit PA 17214). Although I was able to get the copy I read from the local library, it appears that the book can be ordered directly from the publisher. The price on the book was $5.95. *[It's also available from BITS.]*

This is a "how to" book complete with detailed schematics and parts lists. Each chapter is a single project which adds another feature to an ever growing robot. The final project, which the author dubs BUSTER III, can follow a predetermined path laid out with masking tape or "wander" about "exploring" the environment. It also has an override "hunger" control which causes it to seek out its "nest" where it will stay until its batteries are recharged. Once it has "filled itself" it will resume exploring.

Please note that I have not said anything about sight, hearing, arms, legs or humanoid characteristics. That is not what is meant by "robot" in this book. What is meant is a device which consists of complex digital logic (all the circuits are TTL) controlling a simple machine. However, all the basic features of any robot are embodied. "Self" control, "self" protection, an ability to sense the environment, and even a primitive "cry" of "panic" or "hunger" are "programmed" into BUSTER's circuits The underlying principles of control and feedback are thoughtfully applied. The book appears to be a first. And as such it is an excellent book.

One word of reservation: I did find frequent minor errors in the diagrams (unlabeled or mislabeled parts and a few blank entries in some tables), but these errors were all correctable by the reader.

Terry E Weymouth
4702 Beau Bien Ln E
Lisle IL 60532■

ALLEGED MORAL VOID? HARDLY

After settling down and reading the January 1977 "Ask BYTE" column *[page 140]*, I found myself a bit dismayed. As a graduate electrical engineering student, I found your remarks to be accurate and succinct. Basically, I'm concerned about the glaring lack of ethical considerations on Blondefield's and your part.

It has been an unfortunate engineering tradition to say, "If it can be done, let's do it." No questions asked. The big question in engineering (and robotism and artificial intelligence in particular) is usually ignored: "Should we build this?" or "Should computers be given these tasks?" In the case of robots, one should ask what fields of human endeavor would the artificial intelligence of the robot be applied to. Would the robot which is mopping the floor also listen to your problems? (Many people hope so.) The idea here is not to come up with an engineering answer, but to ask which of the possible applications would be detrimental to man. In my opinion, many of the future applications of computers, as envisioned by engineers, are unethical. For instance, Blondefield's naive robot impossible dream seems to be the tip of an ethical void.

Computers are being plugged into applications left and right without an ounce of thought about their adverse impact on society. Examples are "a computer in every home," community information centers, and computer psychiatric counseling. The application of computers affects our view of the world and our fellow man. Their ultimate usage may turn out to be quite an insult. A person who says "I want robots" should look a little deeper and face the humanistic and ethical arguments.

Robert Garner
1795 Stanford Av
Menlo Park CA 94025

Unethical? Detrimental? By what standard do you make your evaluations? By any truly human standard of value, computers and robots are highly desirable technology: the former for their automation of mental drudgery, the latter for their automation of physical drudgery. Human beings are the measure of morality, not the current state of social interaction and technology — which can always be improved. A view of morality which precludes improvement in the quality of life of human beings is a monster which should have been left in the dark ages.■

BYTE's Bits

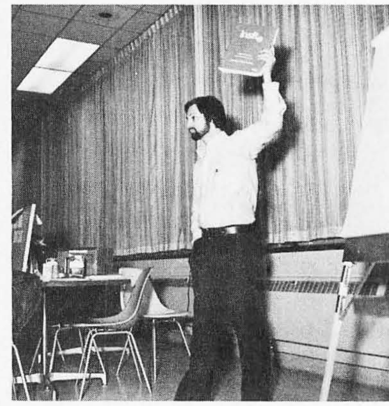


Photo 1: The NECS meeting began with Dave Day, president, attending to formalities (minimal) then introducing the various speakers.

Unbelievable, but It Worked...

Demonstration of the Klugehorn at an NECS meeting...

Notes by C Helmers

I promised Dave Day (newly elected president of NECS) that I'd provide a neat electronic music demonstration at an NECS meeting January 12 1977, to

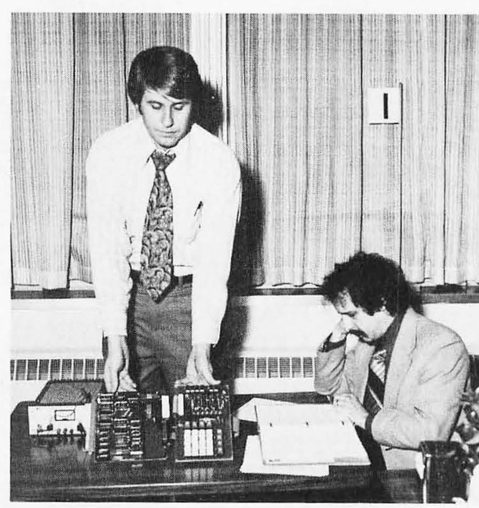


Photo 2: Gary Sawyer of Motorola gave an informative talk about the new Motorola evlauation board style computer, which comes in kit form with a new version of their monitor programs, a Kansas City standard tape interface, and the remote control interactive paddle shown. It features compatibility with larger Motorola development systems, and the ability to expand memory through buffers (not supplied, but wired in via the PC runs). A prototyping area is also present on the card.

be illustrated using my new multichannel orchestral synthesizer which has only now begun to arrive at my lab. But when time for the meeting began approaching, it became apparent that the new synthesizer was not quite going to be ready. What could I do?

Well, two weeks before the event, after having finished the 1200 bps audio interface described in April BYTE, I decided to go all out and prepare a kluge. The result was the "Klugehorn," a single channel music synthesizer peripheral output device which was successfully demonstrated at the meeting on January 12 1977 when a hastily constructed but quite functional music interpreter program played the stored score of most of Scott Joplin's excellent and well known rag, the "Entertainer." (I stopped hand encoding the music in the middle of the last part of the piece at 4:30 AM on the morning of the meeting, so completion was put off until after the demo.)

At about 3:30 in the afternoon on the day of the meeting, I went home from the office with my assistants Dan Pagan and Scott Morrow to take the computer apart, put it in boxes, pack the documentation, and head for Burlington MA and the MITRE Corporation cafeteria where meetings are held. Taking one's processor and system apart for such purposes is a traumatic experience. Any one of an unknowable number of bugs could incubate and come to life, interfering with the successful reconstruction of a working system. Unbelievably, in complete violation of Murphy's law, the system went back together at the MITRE cafeteria, plugged into the wall, and worked without major hitches. (A false read on the tape interface was attributed later to temperature sensitivities, since during the trip to the club meeting it was 0° Fahrenheit and there was ample opportunity for the tape drive and media to be cooled into its nonoperable range.)



Photo 3: Will the stupid program load?

Music is an excellent application of the microprocessor, and even given the limitations of my quick and ad hoc design for a synthesizer, the full range of intelligent real time control was demonstrated. The Klugehorn synthesizer peripheral uses a read only memory controlled divide chain from high frequencies, with 9 bit precision in its approximation to the well tempered scale. (A 4 bit code addresses the read only memory and yields the appropriate one of 12 equally tempered notes of an octave.) This divider output then enters a series of octave dividers selected by a 3 bit code. The resulting pitch is then fed to the shift clock of a 40 bit by 6 bit static shift register, via control logic which also allows the shift register to be loaded by a parallel PIA output port of my computer. The six bits out of the shift register are fed to a 6 bit MC1406 waveform DAC which uses a variable reference voltage created by an 8 bit MC1408 envelope DAC directly connected to another 8 bit PIA port. The analog processing logic of operational amplifiers is used to create an AC signal at the output which has an RMS amplitude set by the envelope and a waveform stored in the shift register. Timing of changes in the envelope amplitude are handled by an interrupt driven algorithm in the processor, which responds to the 4800 Hz clock I normally use for my tape cassette interface. A music interpreter of about 500 bytes in length drives the system. The only major hitch with this particular design as demonstrated was a lack of digitally controlled filters to eliminate the high frequency harmonics of the square edges which result from the step functions used for the waveform and envelopes. The result is a very

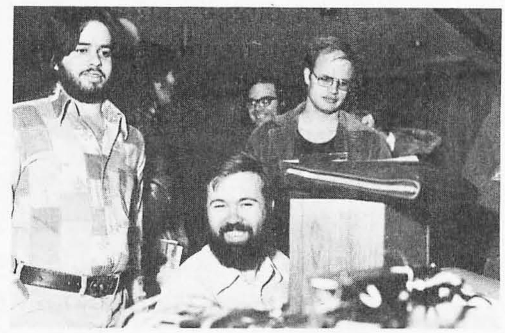


Photo 4: Ah, sweet success. . . .



Photo 5: Explaining how the Klugehorn works, a short tutorial on last minute breadboarding and its results.

flexible synthesis system which can change timbre in several ways: by changing waveforms directly, by changing envelope contours, and by changing the rate at which envelope contours are evolved. More on this later if and when I ever get around to writing an article in BYTE on the subject. (As I write this, a real time music interpreter for the Klugehorn has already been completed; it handles two processes, allows waveform or envelope definitions, simple arithmetic timbre modifications, subroutines and other goodies.)■

A Note on Omega

BYTE readers who are building their own version of the Mini-O Omega receiver system *[described by Ralph Burhans, February 1977 BYTE, page 62; March 1977, page 70, and by Richard Salter, April 1977, page 100]* may be interested in joining the International Omega Association which publishes newsletters and sponsors an annual conference for exchange of information among users and hardware-software builders. This year's conference is expected to include a user workshop and probably will be held this summer on the East Coast. Write for information to: IOA, POB 2324, 1720 EADS St, Arlington VA 22202. Membership in the IOA is open to anyone interested in Omega and VLF navigation problems for an annual fee of $10.

Another organization of interest to navigator types is: The Institute of Navigation, Suite 832, 15th St NW, Washington DC 20005. The ION publishes a quarterly *Journal of the Institute of Navigation (JION)* on all types of navigational subects including Omega, Loran-C, Satellites, Aircraft, Geodesics, etc. Membership dues for the ION are $15 a year which include a subscription to *Navigation.* Several annual conferences are held each year.

Ralph Burhans
EE Dept, Ohio University
Athens OH 45701

Learn About and Take Home a Micro

Wintek has announced the Spring schedule for their workshop, "Hands on Microprocessor Short Course with Free Take Home Microcomputer." Attendees receive a microcomputer to use at the workshop and take home when they leave. The take home computer has a 6800, programmable memory, parallel IO, serial IO, and read only memory with Fantom-II (which supports single step, break points, etc.) and is fully expandable to 65 K. Course schedule is May 10-12, Lafayette IN; May 24-26, Cleveland/Akron OH; June 7-9, Syracuse NY; and June 21-23, Hackensack NJ. Tuition is $495. Wintek Corp, 902 N 9th St, Lafayette IN 47904, (317) 742-6802.■

Continued from page 9

products to date based on a "typical" $42 per 1024 installed bytes. At the bottom of the table is an estimate of the total size of the personal computing retail sales to date which is between 39.5 million and 58.3 million using a conservative estimate of mainframe and peripherals cost. While we (ie: computer enthusiasts and manufacturers who serve them) are not yet as big as the auto industry, we've certainly made a respectable start for the first 21 months of the industry's existence (counting the introduction of the Altair as the origin in time, through the survey point of October 1976). This industry is still expanding as more and more people enter the market both as customers and as suppliers.

Some miscellaneous but interesting results from the survey include responses to questions like "Do you have experience with the assembly or machine language of machine X?", or "What is your experience with high level languages?". Table 2 sorts out the answers to the question of specific processor familiarity based on the total 1448 respondents, and table 3 sorts out the answers to the question of familiarity with various high level languages.

A final result which will be of interest to readers is the nature of interest in specific application goals for small machines. Table 4 shows the results, with rankings. Not surprisingly, the predominant interest was in software development. Why isn't this a surprise? Two reasons come to mind. First is that software is intrinsically fun and challenging for its own sake (read Joseph Weizenbaum's comments on the compulsive programmer, an extreme situation of addiction, in his book *Computer Power and Human Reason*). The second reason is that software development is a characteristic of any activity which uses the computer.

I expect to perform additional surveys for BYTE in the future, since intellectual and business curiosity demand it; we'll see how the field evolves in the future, and document it through this form of statistical measure. But this is enough for now. Copies of the Electro paper are available through the IEEE for those who want more detailed information and and discussion of the survey. This survey was probably the first published which has any direct basis in the actual demographics of personal computers, since its target group comprises the largest known set of such enthusiasts.■

Table 4: Interest in hobby or home applications.

Application	Using	Intend	Serious	Rank
Software development	27%	52%	79%	1
Interactive keyboard games	18.4%	53.3%	71.7%	2
Graphics	6.8%	64.7%	71.5%	3
Real time control (as in household automation.)	6%	59%	65%	4
Personal data bases	8%	54%	62%	5
Games with analog inputs	3%	53%	56%	6
Computer design	12%	42%	54%	7
Educational, excluding games	9.9%	42.3%	52.2%	8
Text processing	9.5%	42.3%	51.8%	9
Non computer hobby	8%	35%	43%	10

Serious: Intend + Using

Table 5: Interest in professional and business applications.

Application	Using	Intend	Serious	Rank
Mathematical or data analysis	27%	34%	61%	1
Simulation or models	15%	33%	48%	2
Professional data base	13%	34%	47%	3
Text processing	12%	30%	42%	4
As part of a communications network	11%	29%	40%	5
Accounting	8%	31%	39%	6
Scheduling, planning	6%	27%	33%	7
Inventory control	6%	26%	32%	8
Mailing lists	6%	23%	29%	9

Serious: Intend + Using

Clubs and Newsletters

Conducted by
Peter Travisano

Utah Computer Association Show

The Utah Computer Association will hold an informational show on Saturday, April 30, 1-5 PM at the Fashion Place Shopping Mall, Salt Lake City UT. There is no admission charge and there will be no selling. The purpose of the show is dissemination of information on personal computing. Club members will be bringing their own systems for discussion and display. For more information call Ernie Dixon, (801) 467-9100, between 6 and 9 PM.

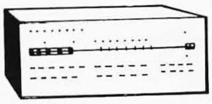
Homebrew Computer Club — Mountain View CA

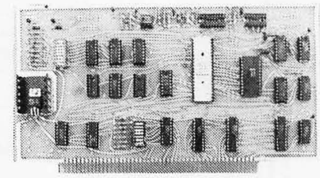
Over the past several months we've featured computer groups which appear to be outstanding: New England Computer Society, Denver Amateur Computer Society and San Diego Computer Society, to name but a few. If one can judge a club by its newsletter, he/she would easily conclude that Homebrew is as its name implies, a unique and innovative club on the vanguard of the computer hobbyist field. Most of the articles are thoughtful, technically accurate and written with consideration for the human as well as the technical side when appropriate.

If you're in the market for a new idea or two, investigate the Homebrew cassette library or subscribe to their newsletter and take advantage of the programs they feature every month. In any case, the Homebrew Computer Club is worth finding out about. Write Homebrew Computer Club, POB 626, Mountain View CA 94042.

Wisconsin Area Computer Society

In looking through the *Homebrew Newsletter*, we came across word of the Wisconsin Area Computer Society. According to *Homebrew* they've been holding meetings for the past 12 months and boast a membership of over 80 people. They're in the process of setting up a software library using paper tapes and cassette tapes. A newsletter is also available. For more information write





Don Stevens, chairman, Wisconsin Area Computer Society, POB 159, Sheboygan Falls WI 53085.

Concord NH Computer Society

Good news for hobbyists in central New Hampshire: A computer group is being formed: the Concord NH Computer Society. For details, contact Bob Vogal, 9 Ridgewood Dr, Concord NH 03301, or phone (603) 224-9402 between 6 and 8 PM.

Santa Barbara Computer Association

The Santa Barbara Computer Assn has its own newsletter. The premier issue was recently printed and it is a good first effort. Like most computer club publications, its focus will be around hardware and software review, interviews with members, meeting announcements and the like. Future meetings will be held the second Wednesday of each month at 8 PM in the Goleta Library, 500 N Fairview, Goleta CA.

To find out more, get in touch with Glenn McComb (805) 968-3112, home, or (805) 967-4410, office, or write POB 2314, Goleta CA 93018.

A Computer Heathkit Users' Group . . .

. . . is being formed. For more information drop a line to CHUG, c/o Charles Floto, 267B Willow St, New Haven CT 06511.

Amateur Computer Group of New Jersey

The Amateur Computer Group of New Jersey is where the action is in the Garden State. Attendance has been well over the 100 mark at main meetings and meetings of special interest groups. This is the group responsible for the Trenton Computer Festival, among other activities.

If you're a computer buff in the Garden State you might look into Amateur Computer Club of New Jersey at UCTI, 1776 Raritan Rd, Scotch Plains NJ 07076.

Triangle Computer Club, Raleigh NC

Hobbyists in the Raleigh area may want to look into the Triangle Amateur Computer Club. The club meets on the last Sunday of the month at Dreyfus Auditorium, located at the Research Triangle Inst, Research Triangle Park, NC. Write Triangle Amateur Computer Club, POB 17523, Raleigh NC 27609.

RAMS – Rochester Area Microcomputer Society

RAMS has been in existence since April 1976, helping members by sponsoring group

purchases, a club library, and coordinating regional computer fests with other area clubs. A newsletter is also available. RAMS is located at POB D, Rochester NY 14609.

Southeastern Michigan Computer Organization

This Detroit based group has grown substantially. Special interest committees catering to individual members needs have been established. One such group is SIGKIM, a small group of avid KIM-1 users and other 6502 based people. In fact, the group has worked out well enough to warrant a monthly appendium to the *Data Bus*. If you are a KIM enthusiast, it might be worth subscribing just to get that information. Anyway, if you're from the Detroit area, it would be worth your while to check out SMCO. Write Southeastern Michigan Computer Organization, POB 9578, N End Station, Detroit MI 48201.

Microprocessor Club of Sidney, Australia

Gary Lynch, a computer buff from down under, was kind enough to send us a clipping from the newspaper, *The Australian,* which carried a story on the initial meeting of the Microprocessor Club of Sydney.

Apparently, there's plenty of interest in computers as evidenced by a fine turnout at this charter meeting. Prospective members should contact Kevin Barns, Electronics Today, Sydney, AUSTRALIA.

Birmingham AL

Birmingham has its own computer group partly due to the Clubs and Newsletters column. Jim Anderson wrote that "as a result of your relaying my note to contact other enthusiasts in the Birmingham area, I received several phone calls and letters and the result is the formation of the Birmingham AL Microprocessor Group."

Meetings are held on the fourth Sunday of the month, with less formal workshop sessions on the second Sunday of the month. At press time there is not a permanent meeting place, so you'll have to get in touch with Jim Anderson at 3905 Shannon Ln, Birmingham AL 35213. (205) 879-9630.

Want to Spread the Word About Your Club or Organization?

Well, that's what the Club and Newsletters column is all about. Put BYTE on your mailing list or send us a letter, and you'll substantially increase your club's visability; simple as that. New groups are especially welcome. ■

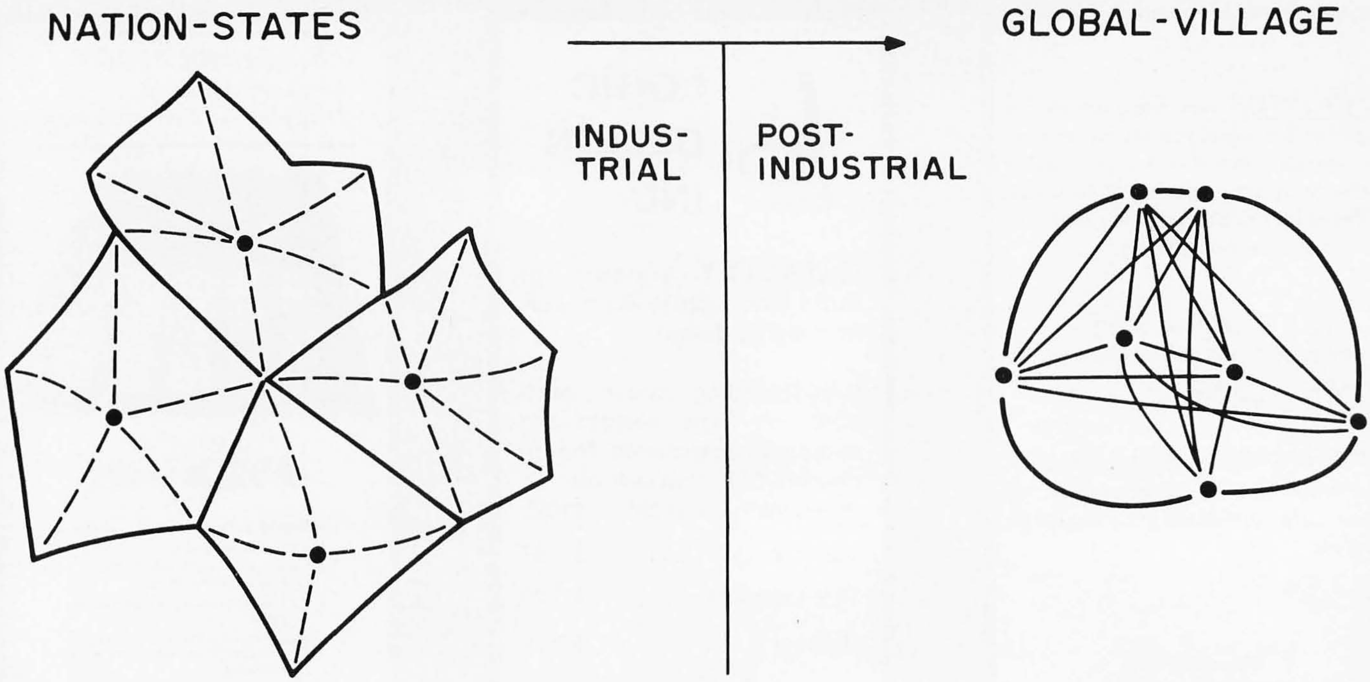


Figure 3: Structures of Governing. In the traditional "nation state" from time immemorial political authority is concentrated in a strong central government. In Roman times, for example, the mode of communication was paper (papyrus) and mechanical transportation (chariot, ships, etc). When the costs of transport are high, the center-margin form of communications is the most cost effective, minimizing the number of paths required. As technology developed, the cost of communications links has declined. Such items as jet transportation, the automobile and electric communications have stressed the decentralized network concept through mass production and lower cost. The result for government might be called the "global village" or even the "network state."

Continued from page 24

1) Golden Ager's Software Co
2) Folsom Prison Software Development Inc
3) Gray Panther Computer Diagnostics
4) Mother Goose Arithmetic Packages Ltd

d. The computer as a system for sale: Putting it all together (software, hardware, application programs, peripherals, maintenance, customer training and documentation), the small business group could sell turnkey operations to customers with more intensive data processing requirements.
   1) Stock inventory system.
   2) Word processing system.
   3) Order processing system.

In general, developing the computer as a product could provide jobs, economic inputs, and self-supporting computer services for the community.

2. Political Power

Marshall McLuhan argues that the speed and capacity of the media (both communications and transportation) determines the form and extent of human associations, notably structures of governing (see figure 3). He contends that the center margin form, illustrated by a powerful central government ruling a large territory, was first actualized in the Roman Empire through the use of papyrus and chariots, and came to its climax in the modern nation state with its mechanized transports like tanks and automobiles. He further contends, however, that the electric media "creates centers everywhere. Margins cease to exist on this planet."[10] More accurate would be to say that centers are *trying* to exist everywhere, that the new speeded-up media are putting enormous stresses on our "mechanically based" forms of organization to make them compatible with centers everywhere.

a. The computer as an extension of central government: It is obvious to many people that, with electric media, authorities (senators, congressmen, cabinet members, etc) no longer need to be located in close physical proximity as they are currently in DC. Conferences, debates, sharing of documents, even back room confidences could be performed using audio or visual or facsimile telecommunications. Such techniques have been developed and are used by corporations with widely dispersed executives participating.

   It is also obvious that citizens no longer need "representatives" sitting in a distant auditorium to vote on their behalf when, through the electric media, they could both be informed of the issues and also vote directly. A "yes" and "no" button in each of our homes would theoretically allow instantly tabulated national referendums on issues that are now voted upon by only a few hundred individuals.

   1) Local administration of federal programs: Line costs for data transmission now exceed other hardware costs, thus discouraging transmission of data from its source and encouraging the establishment of local data bases near the

people the federal program is designed to serve. The uniformity of computer programs from one local data base to another ensures the uniform application of the laws governing the administration of benefits, but avoids the complexities, aloofness, and possible privacy issues of a mammoth centralized data base.

b. The computer as a tool for decentralizing government: While the electric media, including computers, can be used to disperse the administrative and representative functions of central government as proposed in E2a above, this is merely throwing 20th century gadgetry on top of an 18th century form of government. Such proposals are worthwhile in themselves, but they do not tap the serious potential of electric media to decentralize decision-making authority itself.

1) Local administration of neighborhood programs: As the authority to make health, education, welfare, and safety decisions becomes transferred back to the individuals who make up a neighborhood government (through political processes outside the scope of this paper[11]), computer based systems would be used to aid the administration of these programs. Since the kinds of data to be collected from the recipients, record update and purge policies, and the uses of the data are all controlled by a neighborhood government made up of recipients, privacy abuses would be made as difficult as possible.

F. SUMMARY

A few examples of microprocessor-based applications for enjoyment, education, communication, and artistic and political expression have been given. While these and many similar applications will become desirable, and commercially consumable products, home grown applications that result from the merging of technology with human values, or a "liberating" philosophy, will continue to have potentially great impact on the organization of modern life.■

G. NOTES

1. D E Thomsen, "Beating the Game," *Creative Computing*, January-February 1976, pages 22-23.
2. A recent controversial model of world resource and population dynamics can be found in *The Limits To Growth,* D L Meadows, et al, Signet, 1972.
3. E C Berkeley, "Newspeak and Computers: What We Do Depends Upon What We Think," *Computers and People*, pages 6-8, 12, 19.
4. C Johnson, "Span-O-Vision" cartoon, *Creative Computing*, January-February 1976, page 12.
5. M McLuhan, *Understanding Media*, McGraw-Hill, 1965.
6. J Whitney, "Further Reflections of a Culture Savage," *SCCS Interface*, February 1976, pages 22-23.
7. Any introductory psychology book will describe Skinner's major findings. For his view that culture is a Skinner box see *Beyond Freedom and Dignity*, Knopf, 1971.
8. T H Nelson, *Computer Lib/Dream Machines*, 1975. Also "No More Teachers' Dirty Looks," *Computer Decisions*, September 1970, pages 16-23.
9. M Kotler, *Neighborhood Government*, Bobbs-Merrill, 1969.
10. McLuhan, op cit, page 91.
11. Kotler, op cit.

Advanced Microcomputer Products Presents

CONTINENTAL SPECIALTIES CORPORATION

CSC

Prototype Design Equipment

QT SOCKETS AND BUS STRIPS

Universal breadboarding system eliminates the soldering iron. QT sockets provide terminals, each consisting of five pre-connected tie-points, running across the width of the socket. QT bus strips consist of two rows of connected tie-points (in groups of five) which run the length of the bus strip. All QT sockets and bus strips are molded of high-temperature plastic so that leads and wires can be soldered while plugged in. Specially engineered, large-aperture holes guide leads into contacts easily and effortlessly. Contacts are pre-stressed, spring-loaded, non-corrosive, nickel silver alloy. Average contact resistance is 0.5 milliohm initially; 0.4 milliohm after use. Any diameter lead from .015" to .032" may be used. All QT units are .33" thick; all sockets are 1.37" wide; all bus strips are .36" wide. Units mount with 4-40 flat-head screws from the front or 6-32 self-tapping screws from the rear. QT units feature unique snap/lock mechanism which mates units together in seconds to expand or contract a breadboard system at will.

CSC No.	Length, Ins.	Terminals*	Mtg. Dimen., Ins.	List Price
QT SOCKETS				
QT-59S	6.5	118	6.2 x 0.7	$12.50
QT-47S	5.3	94	5.0 x 0.7	10.00
QT-35S	4.1	70	3.8 x 0.7	8.50
QT-18S	2.4	36	2.1 x 0.7	4.75
QT-12S	1.8	24	1.5 x 0.7	3.75
QT-8S	1.4	16	1.1 x 0.7	3.25
QT-7S	1.3	14	1.0 x 0.7	3.00
QT BUS STRIPS				
QT-59B	6.5	20	6.2 x 0.7	$ 2.50
QT-47B	5.3	16	5.0 x 0.7	2.25
QT-35B	4.1	12	3.8 x 0.7	2.00

*Groups of five pre-connected tie-points.

PROTO-BOARD® UNITS

All the speed and convenience of QT Sockets and Bus Strips in both kits and preassembled units. Assemble, test and modify circuits literally as fast as you can think.

PROTO-BOARD 6 KIT

Contains one preassembled QT-47S socket, two preassembled QT-47B bus strips, four 5-way binding posts, metal ground/base plate, non-marring feet and all hardware. Ten minute assembly time. **Size:** 6" l. x 4" w. x 1.4" h. **Weight:** 7 ozs.
CSC Model PB-6 Kit—Complete. List Price $15.95



Now in stock for
Microprocessor Prototyping
NEW SOCKETS & BUS STRIPS

Experimenter 600 . . . . $10.95
.6 spacing for microprocessor applications

Experimentor 300 . . . . $ 9.95
.3 spacing for microprocessor applications

PROTO-BOARD 100 KIT

Contains two preassembled QT-35S sockets, one preassembled QT-35B bus strip, two 5-way binding posts, non-metallic base plate, non-marring feet and all hardware. Ten minute assembly time. **Size:** 4.5" w. x 6" l. x 1.4" h. **Weight:** 7.5 ozs.
CSC Model PB-100 Kit—Complete. List Price $19.95

PROTO-BOARD 101

Fully assembled breadboard contains two QT-35S sockets and four QT-35B bus strips mounted on metal ground/base plate with non-marring feet. Excellent for audio and small digital projects. **Size:** 5.8" l. x 4.5" w. x 1.4" h. **Weight:** 9 ozs.
CSC Model PB-101 Breadboard—Complete. List Price $29.95

PROTO-BOARD 102

Fully assembled breadboard contains two QT-47S sockets, three QT-47B bus strips and one QT-35B bus strip on a metal ground/base plate with non-marring feet. Excellent for intermediate digital needs. **Size:** 7" l. x 4.5" w. x 1.4" h. **Weight:** 10 ozs.
CSC Model PB-102 Breadboard—Complete. List Price $39.95

PROTO-BOARD 103

Fully assembled breadboard contains three QT-59S sockets, four QT-59B and one QT-47B bus strips, four 5-way binding posts on a metal ground/base plate with non-marring feet. Build calculators, interfaces, networks, etc. **Size:** 9" l. x 6" w. x 1.4" h. **Weight:** 1¼ lbs.
CSC Model PB-103 Breadboard—Complete. List Price $59.95

PROTO-BOARD 104

Fully assembled breadboard contains four QT-59S sockets, seven QT-59B bus strips and four 5-way binding posts on a metal ground/base plate with non-marring feet. Build a CPU, encoder, complex display, etc. **Size:** 9.8" l. x 8" w. x 1.4" h. **Weight:** 1½ lbs.
CSC Model PB-104 Breadboard—Complete. List Price $79.95

PROTO-BOARD 203

Fully assembled breadboard contains built-in, short-proof, fused, 5 VDC at 1 amp, regulated power supply, in addition to three QT-59S sockets, four QT-59B bus strips, one QT-47B bus strip and four 5-way binding posts. Capacity for most digital and many analog projects. **Size:** 9.75" l. x 6.5" w. x 3.25" h. **Weight:** 5 lbs.
CSC Model PB-203 Breadboard—Complete. List Price $75.00

PROTO-BOARD 203A

Provides all the features of Proto-Board 203 with additional +15 and −15 VDC at 0.5 amp power supplies with internally adjustable output voltages. **Size:** Same as PB-203. **Weight:** 5.5 lbs.
CSC Model PB-203A Breadboard—Complete. List Price $120.00

LOGIC PROBE LP-1



A compact, versatile design and test tool for digital applications. By simply connecting the clip leads to the circuit power supply, setting the logic family switch to the proper position (TTL/DTL or CMOS) and touching the probe tip to the IC pin in question, the LP-1 will give an instant picture of circuit conditions. Two level detector LED's (HI for logic "1" and LO for logic "0") and a blinking PULSE detector LED display the signal activity. One-shot, low-repetition-rate, narrow pulses are stretched to ⅓ second. A PULSE/MEMORY switch allows one shot events positive or negative transitions to be stored indefinitely. High frequency signals cause the PULSE LED to blink on and off at a 3 Hz rate. When the high frequency signals are unsymmetrical, there is an additional indication. Duty cycles less than 30% activate the LO LED; duty cycles above 70% activate the HI LED. **Input Impedance:** 100,000 ohms. **Threshold:** DTL/TTL, 2.25 V ±0.15 V equals "Logic 1", 0.80 V ±0.01 V equals "Logic 0". HTL/CMOS, 70% of Vcc equals "Logic 1", 30% of Vcc equals "Logic 0". **Minimum Detectable Pulse Width:** 50 nanoseconds. **Input Frequency:** Up to 10 MHz. **Input Overload Protection:** ±50 V continuous; 117 VAC for less than 15 seconds. **Size:** 5.8" l. x 1.0" w. x 0.7" d. **Weight:** 3 ozs. **Power Required:** 30 mA at 5 VDC, 40 mA at 15 VDC, 36 V max. with polarity reversal protection.
CSC Model LP-1 Logic Probe—Complete. List Price $44.95

WE ARE A FULL LINE CSC DISTRIBUTOR. WE CAN SUPPLY ANY PRODUCT OUT OF THE CSC CATALOGUE. IF YOU NEED THEIR LATEST CATALOGUE WRITE AND WE'LL SEND IT ALONG

8K RAM BOARD $198.00

ADVANCED ★ MICROCOMPUTER PRODUCTS

BUILD YOUR OWN COMPLETE COMPUTER SYSTEM



★ AMI EVK 99 6800 BASED MICROCOMPUTER

Ask the Home Brew Computer Club in Santa Clara, California why they choose the EVK 99 Microcomputer. They now have 85 systems. The reasons are clear. Performance, versatility and cost effectiveness. Photo is of expanded EVK 99 Board

Photo of expanded EVK 99
NOW $133.00

1. EVK 99 consists of the following only
   - Qty (1) PCB 10½ x 12
   - (1) 6800 8 bit CPU
   - (1) 6850 ACIA (Asynchronous Communication Interface Adapter)
   - (1) 6820 PIA (Peripheral Interface Adapter)
   - (2) 6830 ROM Subroutine Program Library (1) 6831 16K ROM
   - (4) 6810 1 RAM
   - (1) Operating Manual & Instructions

   To Implement fully expanded system additional IC's are required. (See Photo)
2. UNIVERSAL KLUGE BOARD $98.00
   Completely compatible to the EVK system. This board allows the user to add any additional parts he desires. It has 5400 holes, holds 94 16 pin IC's or others on .2, .4, .6, .9" or greater centers. 2 - 50 pin, 1 - 20 pin flat cable, 1 - 25 pin RS232 connector
3. 16K BYTE RAM BOARD $77.95
   16K x 8 or 8K x 16 organization. Uses 2102's or 21L02's and adapts to slow memory
4. 6 SLOT MOTHER BOARD $29.95
   Connectors mount on .75" centers. May be plugged together for expanded applications
5. EXTENDER BOARD $46.50
   For use with back plane above in card cage configuration.
   New boards are now in development including cassette I/O and others. If you want more information drop us a line

★ MICROPROCESSOR PRODUCTS

8080A/9080A	$24.95	S6860	$15.95			21L02/91L02	1.99
AMI S6800	24.95	S6820	11.00			2102/9102	1.89
AM2901	29.95	S6810 1	6.25			9111/2111	4.75
GI CP1600	69.95	S6850	10.95			9101/2101	5.25
Z80	69.95	S2350	12.95			9112/2112	2.95
6502	29.95	S1883	5.90			7489	1.88
AM2902PC	$ 3.95	S6834 1	22.95			4104/4804	12.95
AM2905PC	10.95	S6800 Programming Manual	15.00			4402A	12.95
AM2906PC	11.95	S6800 Hardware Manual	15.00			4200	12.95
AM2907PC	10.75	[illegible]	22.95			9131APC	14.95
AM2909PC	24.95	[illegible]	175.00			9141APC	14.95
AM2911PC	18.95	8212	$ 4.75	AM2533	7.50	9050/TMS4050	12.95
AM2918	6.75	8224	6.25	AM2401	16.50	9060/TMS4060	12.95
AM29720PC	7.95	8216	4.95	AM1402	6.95	P3101	3.25
AM29721PC	7.95	8226	5.95	AM1403	5.95	5101	11.95
AY 51013A	5.50	8228	9.90	AM1404	6.95	1702A	10.90
AM2812	14.95	9551	19.95	MM500H	2.25	2708	68.00*
AM2841	12.00	9555	19.95	MM5016H	4.95	5204	23.50
AM3341	9.95			AM3347	6.40		

TV GAME KITS

Qty (1) GI AY38500 1 TV Game Chip — ONLY $29.95

TV KIT NO.1 — ONLY $34.95
Includes AY38500 1 TV GameChip, PC Board for video composite output plus instructions. Add $5.95 for 2MHz crystal

TV KIT NO. 2 — ONLY $49.95
Includes AY38500 1 TV GameChip, PC Board, Switches, pots, speaker, crystal, XFMR, Bridge and all parts required for video composite. Works super with terminal. Includes instructions and data sheets

SPECIAL
BUY 10 OF THE FOLLOWING AND YOUR PRICE IS. . .

AMD 8080A............$19.95
AMD 9111....................4.10
AMD 9112....................2.30

RF MODULATOR MODULE

To convert your video composite to RF try UM 1021 Channel 3/4 VHF Modulator — $8.95

- DC Switchable VSWR = 2 max VCC = 6.5 volts
- DC Coupled modulation input with positive transfer characteristic. Designed for use with TV Games

SPECIALS

75452	10/2.50
8T26B	2/5.00
74LS138	1.49
8097	.99
74166	.80
74161	.49
741HC	10/2.50
MH0026CN	3.95
1488	1.29
1489	1.29

★ NEW PRODUCTS

F4702 Baud Rate Gen.	15.95
8214 Priority Int.	11.95
Z80 CPU	49.95
Z80 3881 PIC	9.95
Z80 3882 CPC	9.95
PD 372 Floppy Controller	55.00
PD 371 Mag. Tape Controller	49.95
APP Notes for 372 Floppy	6.95
GI AY5 3550 4½ Digit DVM	29.95
GI AY5 3600 Keyboard Encoder	14.95
Intersil 8038CC Wave Gen.	3.50
AY3507 DVM Chip	12.95
RO3 2513 Uppercase	9.95
RO3 2513 Lowercase	10.95
TMS 4060 NL-2 4K RAM	3.95
MK4096 4K RAM	3.95
1602B UART	5.50
1883 UART	6.50
AY51013A UART	5.50
5101C E 1K CMOS RAM	6.95
WD 1671B Astros	29.95
WD 1941 Dual Baud Gen	9.95
WD 1771 Floppy Disc Cont.	74.95
6330-1 256 Bit OC Prom	2.95
6331-1 256 Bit TS Prom	2.95
6300-1 1K OC Prom	3.50
6301-1 1K TS Prom	3.50
6305-1 2K OC Prom	9.95
6306-1 2K TS Prom	9.95
6340-1 4K OC Prom	18.95
6341-1 4K TS Prom	18.95
6352-1 4K OC Prom	18.95
6353-1 4K TS Prom	18.95

From Iasis David Guzeman's formerly of Intel - Complete 6 Volume Programmed learning course "Microcomputer Design is a Snap" only $99.50
Microcomputer Application Handbook only $7.95
Iasis 8080 Microprocessor System in a book complete instructional and Microcomputer
Send for complete details on all Iasis Products.

Intel MCS8 Users Manual	5.00
Intel MCS4 Users Manual	5.00
Raytheon Linear Catalogue	2.95
Raytheon Quads & Duals Manual	1.95

ALTAIR /IMSAI COMPATIBLE 8K MEMORY

Our new brain board comes complete with 21L02 1 memory chips (500ns) and sockets. All address control and data out lines fully buffered.
$198.00 COMPLETE
If you desire 91L02APC - Please add $9.95

6800 STARTER SET

Only 68 Bucks $68.00
You Get (1) 6800 8 bit CPU
(1) 6820 PIA
(1) 6850 ACIA
(16) 2102 RAM's

8080A STARTER SET

Only $80.00
You Get (1) 8080A 8 bit CPU
(1) 8212 I/O Port
(1) 8224 Clock
(1) 8228 System Controller
(24) 2102 RAM's

All shipments first class or UPS in U.S. or add 5% for shipping and handling. Minimum order $10.00 California Residents add 6% tax. Foreign add 8%.

★ IC MARKET PLACE ★

★ LINEAR

LM105	3.50	LM320K 5.2	1.45	LM381N	1.79
LM106H	2.50	LM320K 12	1.45	LM382N	1.79
LM300H	.80	LM320K 15	1.45	LM555CN	.58
LM301H	.45	LM320T 5	1.75	LM556CN	1.30
LM301CN	.45	LM320T 5.2	1.75	LM567CN	1.45
LM302H	1.25	LM320T 8	1.75	LM565N	1.95
LM304H	1.20	LM320T 12	1.75	LM565H	1.95
LM305H	.85	LM320T 18	1.75	LM703CN	.45
LM305AH	1.25	LM232K 5	9.95	LM709H	.40
LM306N	.95	LM324M	1.65	LM709N	.40
LM307CN	.35	LM339N	1.55	LM710N	.70
LM307H	.35	LM340K 5	1.95	LM711N	.39
LM308H	1.00	LM340K 8	1.95	LM711H	.39
LM308CN	1.00	LM340K 12	1.95	LM723N	.55
LM309H	1.10	LM340K 15	1.95	LM723H	.55
LM309K	1.25	LM340K 18	1.95	LM725H	3.25
LM310CN	1.15	LM340K 24	1.95	LM733N	1.00
LM310H	1.15	LM351CN	.65	LM733CH	1.45
LM311H	.90	LM370N	1.65	LM739N	1.29
LM311CN	.49	LM370H	1.65	LM741CH	.35
LM312H	1.95	LM373N	3.25	LM741CN 14	.39
LM318H	1.50	LM377N	4.00	LM741CD	.49
LM318CN	1.50	LM380N	1.39	LM747H	.75
LM319N	1.30	LM380CN	1.05	LM747N	.75
LM320K 5	1.45			LM748H	.39
				LM748N	.39

LM1414N	1.75	75324	3.50
LM1458CN	.65	SG4501T	2.40
LM1496N	.95	SG4501N	2.40
LM2111N	1.95	SG3524	8.95
LM2901N	1.95	RC4194T	6.95
LM3065N	.69	RC4195T	6.95
LM3900N	.55	RC1458CN	.89
LM3909	1.10	RC1556CN	.99
LM7520	2.25	RC3403AD	2.30
LM7524N	1.75	RC4131CN	.99
LM7525N	1.90	RC4136D	1.90
LM7534N	2.20	RC4558CN	.99
LM7535N	1.25	RC4151CN	6.50

★ CMOS

CD4000	.25	CD4026	1.95	CD4075	.35
CD4001	.25	CD4027	.85	CD4078	.35
CD4002	.25	CD4028	1.50	CD4081	.35
CD4006	1.75	CD4029	1.99	CD4082	.35
CD4007	.25	CD4030	.60	CD4510	1.95
CD4008	1.95	CD4033	2.49	CD4511	1.40
CD4009	.48	CD4034	4.95	CD4516	1.95
CD4010	.48	CD4035	1.79	CD4518	2.45
CD4011	.28	CD4040	2.10	CD4520	2.49
CD4012	.28	CD4042	1.45	CD4522	2.56
CD4013	.60	CD4043	1.45	CD4526	2.56
CD4014	1.95	CD4044	1.95	CD4527	2.70
CD4015	1.95	CD4049	.61	CD4528	1.50
CD4016	.59	CD4050	.61	CD4531	1.99
CD4017	1.25	CD4051	2.25	CD4543	2.25
CD4018	2.39	CD4066	1.20	CD4555	1.75
CD4019	1.95	CD4068	.35	CD4556	1.75
CD4020	1.35	CD4069	.35	CD4581	4.95
CD4021	1.85	CD4070	.99	CD4582	1.95
CD4023	.29	CD4071	.35	CD4585	2.45
CD4024	1.29	CD4072	.35	Also	
CD4025	.29	CD4073	.35	5101	$11.95

★ 7400 TTL

7400N	.18	7445N	.99	74122N	.44	74185N	$2.10
7401N	.18	7446N	.99	74123N	.60	74190N	$1.30
7402N	.18	7447N	.99	74125N	.79	74191N	$1.30
7403N	.18	7448N	.99	74126N	.79	74192N	$1.10
7404N	.19	7450N	.26	74132N	.70	74193N	$1.10
7405N	.20	7451N	.26	74141N	.88	74194N	.95
7406N	.25	7453N	.26	74145N	.70	74195N	.95
7407N	.25	7454N	.26	74147N	$1.59	74196N	.95
7408N	.21	7470N	.45	74148N	$1.29	74197N	.95
7409N	.21	7472N	.30	74150N	$1.15	74198N	$1.45
7410N	.21	7473N	.30	74151N	.98	74199N	$1.85
7411N	.21	7474N	.30	74153N	.98	74251N	$1.45
7412N	.21	7475N	.69	74154N	$1.15	74279N	$1.10
7413N	.45	7476N	.42	74155N	.95	8093N	.49
7414N	.69	7480N	.70	74156N	.95	8094N	.49
7416N	.35	7482N	.95	74157N	.95	8095N	.99
7417N	.35	7483N	.95	74160N	.99	8096N	.99
7420N	.19	7485N	.89	74161N	.99	8097N	.99
7421N	.36	7486N	.28	74162N	.99	8098N	.99
7423N	.35	7489N	$1.88	74163N	.99	75150N	$1.70
7425N	.36	7490N	.60	74164N	.99	75450N	$1.10
7426N	.35	7491N	.70	74165N	.99	75451N	.59
7427N	.35	7492N	.59	74166N	$1.10	75452N	.59
7428N	.28	7493N	.79	74170N	$4.20	75453N	.59
7430N	.21	7494N	.79	74172N	$16.25	75454N	.59
7432N	.31	7495N	.79	74173N	$1.40	75491N	.70
7433N	.30	7496N	.79	74174N	$1.25	75492N	.70
7437N	.35	7497N	$3.75	74175N	$1.25	75493N	$1.10
7438N	.35	74100N	$1.10	74176N	.79	75494N	$1.10
7440N	.21	74104N	.53	74177N	.79		
7441N	$1.05	74105N	.53	74180N	.79		
7442N	.69	74107N	.33	74181N	$2.75		
7443N	.69	74109N	.33	74182N	.79		
7444N	$1.50	74121N	.35	74184N	$2.10		

100 PIECES MIX TAKE 10% DISCOUNT
1000 PIECES MIX TAKE 15% DISCOUNT

PLESSEY SEMICONDUCTORS

RADIO COMMUNICATION

SL1610C	1.95	SL1624C	2.75
SL1611C	1.95	SL1630C	1.95
SL1612C	1.95	SL1640C	1.95
SL1613C	2.40	SL1641C	6.95
SL1620C	2.55	SL1645C	5.95
SL1621C	2.95	SL1650C	5.75
SL1622C	2.95	SL1651C	4.95
SL1623C	2.95	SL1680C	3.75

Application Manual for SL600 Circuits 1.95
Plessey Semiconductor Catalogue 3.95

DIVIDERS

SP8600B	6.95	SP8620B	12.50	SP8640B	6.20
SP8601B	4.25	SP8621B	8.50	PS8641B	7.50
SP8602B	16.00	SP8622B	6.50	SP8642B	12.50
SP8603B	14.50	SP8630B	20.00	SP8643B	20.00
SP8604B	8.25	SP8631B	12.00	SP8646B	6.50
SP8607B	17.00	SP8632B	10.00	SP8647B	8.00
SP8613B	17.50	SP8634B	35.00	SP8650B	25.00
SP8614B	18.00	SP8635B	27.50	SP8651B	18.50
SP8615B	20.00	SP8638B	22.50	SP8652B	12.00
SP8616B	22.00	SP8637B	18.00		

ADVANCED ★ MICROCOMPUTER PRODUCTS
P. O. BOX 17329
IRVINE, CALIFORNIA 92713
PHONE (714) 968-3655

★ ORDER YOUR NEW CATALOG NOW!

Z-80 $49.95 4K STATIC RAM ★ 2708 $49.95

WIREWRAP SOCKETS - GOLD

14 Pin WW 1 level	.30
14 Pin WW 2 level	.33
14 Pin WW 3 level	.37
16 Pin WW 1 level	.31
16 Pin WW 2 level	.34
16 Pin WW 3 level	.38
18 Pin WW 3 level	.60
24 Pin WW 1 level	.70
24 Pin WW 2 level	.75
24 Pin WW 3 level	.85
40 Pin WW 3 level	1.25

EDGE CARD CONNECTORS

IMSA1 S100 WW .250" Row Spacing	$5.00
IMSA1 S100 ST .250" Row Spacing	5.00
Altair S100 WW Wire Wrap	5.50
Altair S100 ST Soldertail	5.25

D-SUBMIN CONNECTORS

RS-23225 Pin Male Part No. DB25P	$3.50
RS-23225 Pin Female Part No. DB25S	3.95
RS-232 Plastic Hood for above	.75

CRYSTALS

Microprocessors Timebases
TV Games Precision Crystals

Frequency	Case	Price	Frequency	Case	Price
1.000 MHz	HC6/HC33	4.95	13.000 MHz	HC18	4.95
2.000 MHz	HC6/HC33	5.85	14.31818 MHz	HC18	4.95
2.097162MHz	HC6	6.95	18.000 MHz	HC18	4.95
4.000 MHz	HC18	4.95	20.000 MHz	HC18	4.95
5.000 MHz	HC18	4.95	32.000 MHz	HC18	4.95
6.000 MHz	HC18	4.95	100Kc	HC13	12.95
10.000 MHz	HC18	4.95	3579.45 Kc	HC18	4.95
			2.4576 MHz		5.95

★ Exar

★ FUNCTION GENERATOR KIT ★

The Function Generator Kit features sine, triangle and square wave, THD 0.5% typ., AM/FM capability.

XR 2206KA — Includes monolithic function generator IC, PC board, and assembly instruction manual — 18.95

★ TIMING CIRCUITS ★

XR555CP	Monolithic Timer	.90
XR556CP	Dual 555 Timer	1.75
XR2556CP	Dual Timing Circuit	2.95
XR2240CP	Programmable Timer	4.20
XR320P	Timer	1.40

★ COMMUNICATION CIRCUITS ★

XR1488	Quad RS232 Driver	1.45
XR1489	Quad RS232 Receiver	1.45
XR2208	Four Quadrant Multiplier	4.75

★ PHASE LOCKED LOOPS ★

XR2211CP	FSK Demod/Tone Decoder	5.95
XR567CP	Tone Decoder	1.80
XR25567CP	Tone Decoder	1.80

★ STEREO DECODERS ★

XR1310P	FM Stereo Demodulator	3.00
XR1310EP	FM Stereo Demodulator	3.00
XR1800	Stereo Demodulator	3.00

★ FUNCTION GENERATORS ★

XR2206CP	Monolithic Function Generator	5.25
XR2207CP	Current Controlled Oscillator	3.75

★ OPERATIONAL AMPLIFIERS ★

XR742CP	Zero Crossing AC Trigger	3.00
XR4558CP	Dual Hi Gain Op Amp	1.20
XR4136CP	Quad Op Amp	1.75
XR4202CP	Programmable Quad Op Amp	4.25

2708 PRIME — ONLY $49.95

Build Your Own MONITOR

Now in Stock — NEW

M3000 - 100 12-inch display module	$229.00
M2000 - 155 9-inch display module	$208.00

*Add $10.00 Shipping

MOTOROLA Exorciser, MEK6800DI and MEK6800D2 Compatible Modules

9601 16 Slot mother board - Inc. Conn.	$175.00
9602 16 Slot card cage 19" rack mount	$ 72.00
9610 Utility Prototyping Board	$ 36.00
9615 4K Eprom Module (1702A)	$350.00
9620 Parallel I/O Module	$375.00
9626 8K Static Ram Module	$350.00
9650 8 Channel Duplex Ansyn I/O	$390.00

All assembled and tested. Not Kits

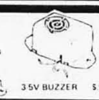
★ MISCELLANY

RIBBON CABLE
8 Conductor

1 - 9 ft	10 - 49 ft	50ft up
.35/ft.	.25	.23



ELCO VARIPAK 2 H
2 Slot cardholders
4/$1.00



3.5V BUZZER $.95

LED'S
- .185" Dia (Red) 5/$1.00
- .185" Dia (Green) 4/$1.00
- .185" Dia (Yellow) 4/$1.00

KEYBOARD



63 key array, terminal keyboard, unencoded with high quality switches. Black w/white legends. Special Offer: Buy now at $19.95 and we will give you a free calculator type keyboard

AY52376 Keyboard Encoder with keyboard Only $14.95



TOUCHTONE KEYBOARD FOR TELEPHONE
Use with MK5085 N integrated tone dialer & 3.579 Hz crystal. Crystal available for $4.95
Only $9.95

NOW!
A 19 key Number Keyboard compatible with our 63 Key Keyboard. Same manufacture and design. Keys for 0 to 9 and 110,300, FDX/HDX, minus Auto LF, Line Loc, Alt Char set and Char set lock.
ONLY $11.95 FIRST TIME OFFERED
(Identical to our 63 Key as pictured except 19 keys.)

★ CLOCK CHIPS

MM5311	24 Pin BCD Output 6 Digits	$4.50
MM5313	28 Pin 6 Digits	4.00
MM5314	24 Pin 6 Digits	3.50
MM5316	[illegible] 40 Pin Alarm Ledout	4.50
MM5375	Clock Chip	3.50

★ IC SOCKETS

Low Profile Tinplate

8 Pin	.17	24 Pin	.41
14 Pin	.20	28 Pin	.49
16 Pin	.22	40 Pin	.63
18 Pin	.31	22 Pin	.35

★ RESISTORS

Carbon Resistors 1/4 Watt ±5%
Most Standard Resistance Valves
NOTE: Multiples of 5 pcs per valve minimum

Qty (each valve)	Price
5-20	.05 each
25-50	.045 each
55-100	.04 each
105-500	.03 each
505 up	.025 each

★ VARIABLE RESISTORS

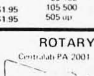
Centralab RN4 NAYSD503A (HMP 50K) $1.95
RN4 NAYSD503A (53C3 50K S) $1.95

★ CORCOM EMI FILTER

- CORCOM 2B4 2A • Only $4.95
  115/230V 50-400Hz Test 2100VDC
- CORNELL DUBILIER NF21015
  2x4A 250 VAC Only $4.95
- DIP SWITCH
  8 SPST Slide Switches in Dip $2.25

ROTARY SWITCH

Centralab PA 2001 $1.95



C&K V 25 Bat handle toggle switch $1.50

GODBOUT

BILL GODBOUT ELECTRONICS
BOX 2355, OAKLAND AIRPORT, CA 94614

TERMS: Add 50¢ to orders under $10; over $10 add 5% for shipping & handling. BankAmericard® /Mastercharge®: call (415) 562-0636, 24 hours. Cal res add tax. No CODs.

SPDT switch



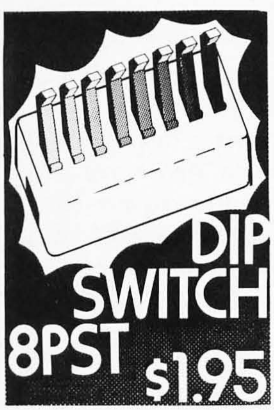
95¢

This beautiful, brand new SPDT miniature switch is designed for PC board mounting (bushing not threaded). Includes mounting frame for extra support.

PC MOUNT......ORDER #SW-3

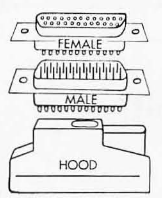
ALSO AVAILABLE: Standard miniature SPDT toggle switch, w/ all mounting hardware. **$1.15** each or **5/$5.00**

STANDARD......ORDER #SW-2





LOOKING FOR A GOOD CONNECTION?



25 PIN RS-232 CONNECTORS: submini D type.

Male plug with plastic hood, part #DB25P $3.95

Female jack, part #DB25S $3.95

74LS TTL

74LS00	$0.36	74LS139	1.38
74LS01	0.36	74LS155	1.38
74LS02	0.36	74LS157	1.25
74LS04	0.42	74LS160	1.85
74LS08	0.38	74LS161	1.85
74LS10	0.36	74LS162	1.85
74LS11	0.38	74LS163	1.85
74LS14	1.38	74LS168	1.87
74LS20	0.36	74LS169	1.87
74LS21	0.38	74LS174	1.38
74LS22	0.38	74LS175	1.35
74LS27	0.38	74LS221	1.38
74LS30	0.36	74LS240	1.88
74LS32	0.38	74LS257	1.25
74LS37	0.53	74LS258	1.38
74LS38	0.53	74LS273	2.25
74LS42	1.25	74LS283	1.20
74LS74	0.56	74LS367	1.00
74LS75	0.85	74LS368	1.00
74LS109	0.60	74LS377	1.88
74LS124	2.50	74LS378	1.38
74LS125	0.75	81LS95	1.13
74LS126	0.75	81LS96	1.13
74LS132	1.50	81LS97	1.13
74LS138	1.38	81LS98	1.13

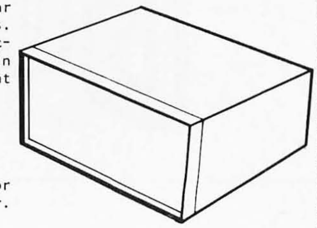
PLAY WITH WIRE?

SEND FOR FREE FLYER!

VECTOR-PAK enclosures

This is not another painted box, but is equivalent to the fancy enclosures used for industrial and computer oriented equipment. Tasteful matte color scheme and extensive use of brushed aluminum add to the beauty. It's shipped knocked down for easy drilling, machining, or labelling, and is easily assembled: now you can wrap your project up in style.

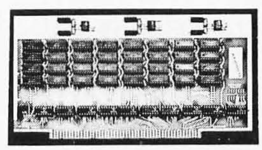
There are no screws or fasteners to mar the lines of these beautiful enclosures. Has provisions for card guides, connectors, etc. on the inside. Available in black or computer blue with white front panel; shipped unassembled.



#VP5-17-17U 5.51"H, 17.58"W, 17.1"D **$79.25**

#VP7-17-17U 7.26"H, 17.58"W, 17.1"D **$84.00**

#VP9-17-17U 9.01"H, 17.58"W, 21.6"D or exact same size as IMSAI microcomputer. **$96.50**

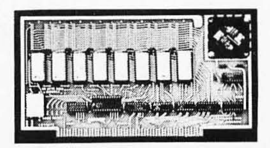


ECONORAM

KIT: $100 WIRED: $130

4K X 8, S-100 buss compatible full feature memory. Guaranteed faster than 500 ns, current draw under 750 mA...both specs over full temp range.

Fully buffered; DIPSWITCH address selector; sockets included for all ICs; uses lo-power Schottky support ICs: the best combination of value and economy.



ECONOROM

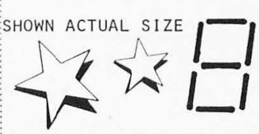
Our family of ECONOROM boards offers 3 different capacities of eraseable ROM... suitable for holding any program or routine you want. Similar to ECONORAM, and also very low power: 5V @ ½A, -12V @ 150 mA.

2K X 8 BOARD.............$135
4K X 8 BOARD.............$180
8K X 8 BOARD.............$270
4K 8080 SOFTWARE BOARD...$190

ALL PRICES ARE FOR KIT FORM

A BIG, BRIGHT .5" DIGIT

SHOWN ACTUAL SIZE



READOUT

FND510 COM. ANODE...$0.95
FND503 COM. CATHODE $0.95
10 OF ONE KIND......$8.50

SPECIAL!

VECTOR'S "SLIT-N-WRAP"

$24.50
+2 lbs postage

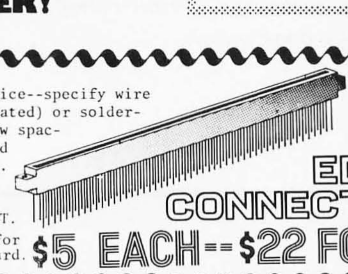


This tool is a manual wrapping device, which supplies insulated wire directly to wrapping posts without pre-stripping or pre-cutting. More than one level of wrap is rarely required. Speed your breadboarding time tremendously with this tool, and don't bother stocking all that pre-stripped and pre-cut wire you would normally require.

100 PIN EDGE CONNECTOR

Now you have a choice--specify wire wrap pins (illustrated) or soldertail with .250" row spacing. For IMSAI and Altair peripherals. Wire wrap part # S-100WW. Soldertail part # S-100ST.

#S-100ST is ideal for the IMSAI motherboard.



$5 EACH--$22 FOR 5

sockets

LOW PROFILE Soldertail, tin plated		WIRE WRAP 3 level, gold plated	
14 pin	10/$1.95	14 pin	10/$3.70
16 pin	10/$2.15	16 pin	10/$3.85
18 pin	10/$2.75	18 pin	1/$0.75
20 pin	10/$3.10	22 pin	1/$1.50
22 pin	10/$3.50	24 pin	1/$1.00
24 pin	10/$3.60	28 pin	1/$1.25
36 pin	10/$5.50	36 pin	1/$1.35
40 pin	10/$6.15	40 pin	1/$1.75

from VECTOR!

This is one of the neatest Altair accessories we've seen. It accepts virtually any size IC package, has a power and ground plane on opposite sides of the board for extra capacitance. Room for 4 regulators, 1 heat sink provided with board. By the way, the sockets are shown only for illustration, but they get the point across that you can stuff a lot of ICs on here --- implement your own memory boards, I/O boards, etc.



UNIVERSAL PROTOTYPE BOARD ➡ ➡ ➡ $19.95

CRYSTALS
THESE FREQUENCIES ONLY

Part #	Frequency	Case/Style	Price
CY1A	1.000 MHz	HC33/U	$5.95
CY2A	2.000 MHz	HC33/U	$5.95
CY3A	4.000 MHz	HC18/U	$4.95
CY7A	5.000 MHz	HC18/U	$4.95
CY12A	10.000 MHz	HC18/U	$4.95
CY14A	14.31818 MHz	HC18U	$4.95
CY19A	18.000 MHz	HC18/U	$4.95
CY22A	20.000 MHz	HC18/U	$4.95
CY30B	32.000 MHz	HC18/U	$4.95

XR-2206KB Kit $27.95 **XR-2206KA Kit $17.95**

EXAR

WAVEFORM GENERATORS	
XR-205	$8.40
XR-2206CP	4.49
XR-2207CP	3.85

STEREO DECODERS	
XR-1310CP	$3.20
XR-1310EP	3.20
XR-1800P	3.20
XR-2567	2.99

MISCELLANEOUS	
XR-2211CP	$6.70
XR-4136	.99
XR-1468	3.85
XR-1488	5.80
XR-1489	4.80
XR-2208	5.20

TIMERS	
XR-555CP	$ .9
XR-320P	1.55
XR-556CP	1.85
XR-2556CP	3.20
XR-2240CP	3.25

PHASE LOCKED LOOPS	
XR-210	5.20
XR-215	6.60
XR-567CP	1.95
XR-567CT	1.70

CONNECTORS
PRINTED CIRCUIT EDGE-CARD
.156 Spacing-Tin-Double Read-Out
Bifurcated Contacts — Fits .054 to .070 P.C. Cards

15/30	PINS (Solder Eyelet)	$1.95
18/36	PINS (Solder Eyelet)	$2.49
22/44	PINS (Solder Eyelet)	$2.95
50/100 (.100 Spacing)	PINS (Solder Eyelet)	$6.95

25 PIN-D SUBMINATURE

DB25P	PLUG	$3.25
DB25S	SOCKET	$4.95

iMC 3½ DIGIT DVM KIT



This 0-2 VDC .05 per cent digital voltmeter features the Motorola 3½ digit DVM chip set. It has a .4" LED display and operates from a single +5V power supply. The unit is provided complete with an injection molded black plastic case complete with Bezel. An optional power supply is available which fits into the same case as the 0-2V DVM allowing 117 VAC operation.

A. 0-2V DVM with Case $49.95
B. 5V Power Supply $14.95

VECTOR WIRING PENCIL
Vector Wiring Pencil P173 consists of a hand held featherweight (under one ounce) tool which is used to guide and wrap insulated wire, fed off a self-contained replaceable bobbin, onto component leads or terminals installed on pre-punched "P" Pattern Vectorbord. Connections between the wrapped wire and component leads, pads or terminals are made by soldering. Complete with 250 FT of red wire. **$9.50**

REPLACEMENT WIRE — BOBBINS FOR WIRING PENCIL

W36-3-A-Pkg. 3	250 ft. 36 AWG GREEN	$2.40
W36-3-B-Pkg. 3	250 ft. 36 AWG RED	$2.40
W36-3-C-Pkg. 3	250 ft. 36 AWG CLEAR	$2.40
W36-3-D-Pkg. 3	250 ft. 36 AWG BLUE	$2.40

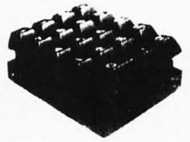
1/16 VECTOR BOARD

	0.1" Hole Spacing Part No.	P-Pattern L	W	Price 1	2-Up
PHENOLIC	64P44 062XXXP	4.50	6.50	1.72	1.54
	169P44 02XXXP	4.50	17.00	3.69	3.32
EPOXY GLASS	64P44 062	4.50	6.50	2.07	1.86
	84P44 062	4.50	8.50	2.56	2.31
	169P44 062	4.50	17.00	5.04	4.53
	169P84 062	8.50	17.00	9.23	8.26
EPOXY GLASS COPPER CLAD	169P44 062C1	4.50	17.00	6.80	6.12

HEAT SINKS

205-CB	Beryllium Copper Heat Sink with Black Finish for TO-5	$ .25
291-.36H	Aluminum Heat Sink for TO-220 Transistors & Regulators	$ .25
680-.75A	Black Anodized Aluminum	$1.60

HEXADECIMAL ENCODER 19-KEY PAD



- 1 - 0
- ABCDEF
- Return Key
- Optional Key (Period)
- — Key

$10.95 each

63 KEY KEYBOARD

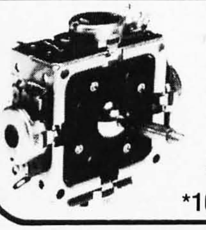


This keyboard features 63 unencoded SPST keys, unattached to any kind of P.C.B. A very solid molded plastic 13" x 4" base suits most applications.

$19.95

HD0165 16 LINE TO FOUR BIT PARALLEL KEYBOARD ENCODER $7.95

JOYSTICK



These joysticks feature four * potentiometers, that vary resistance proportional to the angle of the stick. Sturdy metal construction with plastics components only at the movable joint. Perfect for electronic games and instrumentation.

***5K Pots $6.95**
***100K Pots $7.95**

MICROPROCESSOR COMPONENTS

8080A	CPU	$19.95
8212	8 Bit Input/Output	4.95
8214	Priority Interrupt Control	15.95
8216	Bi-Directional Bus Driver	6.95
8224	Clock Generator/Driver	10.95
CDP1802	- with user manual	39.95
8228	System Controller - Bus Driver	$10.95
MC6800L	8 Bit MPU	35.00
MC6820L	Periph. Interface Adapter	15.00
MC6810AP1	128 x 8 Static RAM	6.00
MC6830L7	1024 x 8 Bit ROM	18.00
Z80	CPU	49.95

CPU'S

8080	Super 8008	24.95
8080A	Super 8008	19.95

SR'S

2504	1024 Dynamic	$ 3.95
2518	Hex 32 BIT	7.00
2519	Hex 40 BIT	4.00
2524	512 Dynamic	2.49
2525	1024 Dynamic	6.00
2527	Dual 256 BIT	3.95
2529	Dual 512 BIT	4.00
2532	Quad 80 BIT	3.95
2533	1024 Static	7.95
3341	Fifo	6.95
74LS670	16 x 4 Reg	3.95

UART'S

AY-5-1013	30K Baud	$5.95

ROM'S

2513	Char. Gen.	$ 9.95
2516	Char. Gen.	10.95
74S387	1024-Bit Programmable	1.95

RAM'S

1101	256 x 1	Static	$ 1.49
2101	256 x 4	Static	5.95
2102	1024 x 1	Static	1.75
2107/5280	4096 x 1	Dynamic	4.95
2111	256 x 4	Static	6.95
7489	16 x 4	Static	2.49
8101	256 x 4	Static	6.95
8111	256 x 4	Static	6.95
8599	16 x 4	Static	3.49
91L02	1024 x 1	Static	2.25
74200	256 x 1	Static	6.95
93421	256 x 1	Static	2.95
MM5262	2K x 1	Dynamic	2 for 1.00

PROMS

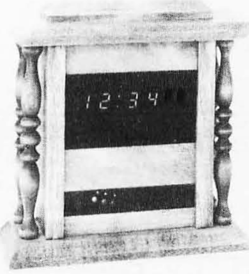
1702A	2048	Famos	$ 9.95
5203	2048	Famos	14.95
82S23	32 x 8	Open C	5.00
82S123	32 x 8	Tristate	5.00
74S287	1024	Static	7.95
3601	256 x 4	Fast	3.95
2708	8K	Eprom	34.95
6301-1	1024	Tri-State Bipolar	3.49
6330-1	256	Open Collector Bipolar	2.95
6331-1	256	Tri-State Bipolar	2.95

SPECIAL REQUESTED ITEMS

MC3061P	3.50	CD4508	6.75	82S115	25.00	3341	6.95
MC4016P (74416)	7.50	CD4515	6.50	5841	9.95	9368	3.95
MC14583	3.50	CD4520	2.70	MK50240	17.50	MC1408L7	9.95
MC14562	14.50	MCM6571	17.50	11C90	19.95	LD110/LD111	25.00/set
CD4059	9.95	MCM6574	17.50	DS0026CH	3.75	AY-5-9100	17.50 ea.
CD4070	.95	MCM6575	17.50	TIL308	10.50	95H90	13.95

ELECTRONIC 'PENDULUM' CLOCK

- Unique Electronic 'Swing' Pendulum
- Real Furniture Quality
- Handicrafted Unfinished Wood Case
- Large .7" Hours, and Minutes LED Display
- Alarm Feature
- 12 or 24 Hour Mode
- Time Set Push Buttons
- Use Module For Easy Assembly
- **Special programed ROM in pendulum for animated effect!**



Dimensions - 10¼" x 10½" x 3¼"

KIT — UNFINISHED* $59.95
*Case Unassembled

ASSEMBLED — STAINED* $69.95
*Case Assembled

BUGBOOK ®
Continuing Education Series



BUGBOOK I & II - Basic concepts of TTL Logic — over 90 experiments **$17.00/set**

BUGBOOK IIa - Introduces UART — recommended for RTTY enthusiast **$5.00/book**

BUGBOOK III - Explores 8080 chip — introduces Mark 80 Microcomputer **$15.00/book**

555 TIMER APPLICATIONS SOURCEBOOK WITH EXPERIMENTS — over 100 design techniques **$6.95/book**

CMOS-M-DESIGNERS PRIMER AND HANDBOOK
a complete CMOS instruction manual **$6.00**

Introductory Offer — all 6 books (worth $49.95)
SPECIAL - $42.95

CONTINENTAL SPECIALTIES

PROTO BOARD 6
$15.95
(6" long X 4" wide)



Other CS Proto Boards

PB100	- 4.5" x 6"	$ 19.95
PB101	- 5.8" x 4.5"	29.95
PB102	- 7" x 4.5"	39.95
PB103	- 9" x 6"	59.95
PB104	- 9.5" x 8"	79.95
PB203	- 9.75 x 6½ x 2¾	75.00
PB203A	- 9.75 x 6½ x 2¾ (includes power supply)	120.00

Logic Monitor $84.95
for DTL, HTL, TTL or CMOS Devices

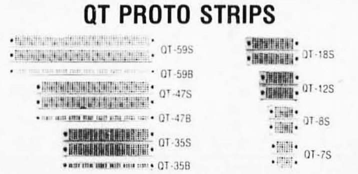
PROTO CLIPS

14 PIN	$4.50
16 PIN	4.75
24 PIN	8.50

DESIGN MATES

DM1 - Circuit Designer 54.95
DM2 - Function Generator 69.95
DM3 - RC Bridge 59.95

QT PROTO STRIPS



QT type	#holes	price
QT-59S	590	12.50
QT-59B	bus strip	2.50
QT-47S	470	10.00
QT-47B	bus strip	2.25
QT-35S	350	8.50
QT-35B	bus strip	2.00
QT-18S	180	4.75
QT-12S	120	3.75
QT-8S	80	3.25
QT-7S	70	3.00

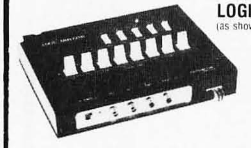
Experimentor 300	$ 9.95
Experimentor 600	$10.95

$5.00 Minimum Order — U.S. Funds Only
California Residents — Add 6% Sales Tax
Spec Sheets - 25¢ — Send 35¢ Stamp for 1977A Catalog
Dealer Discount Available — Request Pricing

NOW OPEN SATURDAYS

James ELECTRONICS
1021-A HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097
All Advertised Prices Good Thru May

PARATRONICS
LOGIC ANALYZER KIT
(as shown in Feb. Popular Electronics)



Troubleshoot any type of digital system including Microcomputers.
(includes IC, power supply, PC Board, case and manual)
LA100 KIT $189.00

DIGITAL AUTO INSTRUMENT
SEVEN DIFFERENT INSTRUMENTS!
MEETS OR EXCEEDS ORIGINAL AUTOMOTIVE SPECS.
*Please specify which **one** of the seven models you want when ordering - these **do not** all come in one unit. Each model must be bought separately.*

1. TACHOMETER 0-9900 RPM 4, 6 or 8 Cyclinders
2. WATER TEMP. 100-250° F
3. FUEL LEVEL Percentage Low Fuel Indicator
4. SPEEDOMETER* 0-99 MPH
5. OIL PRESSURE 0-80 PSI
6. OIL TEMP 200-350° F
7. BATTERY MONITOR ±1 Volts From 11-15 VDC



BRIGHT YELLOW ORANGE .3" LED DISPLAY!
Kit includes case, bracket and all componets — complete. Nothing else to buy! 12 Volt NEG GRD.
DIMENSIONS: 4½ x 4 x 2 **KIT: $49.95**
* Add $10.00 for required speed transducer. **ASSEMBLED: $59.95**

DIGITAL STOPWATCH
- Bright 6 Digit LED Display
- Times to 59 minutes 59.59 seconds
- Crystal Controlled Time Base
- Three Stopwatches in One
  Times Single Event — Split & Taylor
- Size 4.5" x 2.15" x .90" (4½ ounces)
- Uses 3 Penlite Cells



Kit — $39.95
Assembled — $49.95
Heavy Duty Carry Case $5.95
Stop Watch Chip Only (7205) $19.95

Timeband™
A Trademark of Fairchild Camera and Instrument Corporation

DIGITAL ALARM CLOCK $16.95
- 24-Hour Alarm
- "DOZE" Button
- 100% Solid State
- Large Red Led Display (.8" high)
- AM/PM Indicator
- Seconds Display at touch of button
- SPECIFY BLACK OR IVORY



QUARTZ DIGITAL AUTO CLOCK OR ELAPSED TIMER!
Elapsed Timer: Hrs, Mins and Secs
12 or 24 Hr Capacity
Simple Reset - Start Pushbutton Control

Complete kit includes mounting bracket, case and all components, nothing else to buy. Features MM5314 chip. Large .4" LED's. Accuracy better than ± min. per mo. internal battery backup. 12 volt non-polar operation.



DIMENSIONS 4½" x 4" x 2"
12 or 24 HOUR MODE

Kit: $29.95
Assembled: $39.95

CASE ONLY (includes hardware, mounting bracket and bezel) **$6.50**

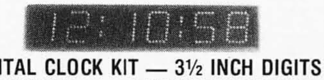
JE700 CLOCK



12 or 24 Hour
115 VAC

The JE700 is a low cost digital clock, but is a very high quality unit. The unit features a simulated walnut case with dimensions of 6" x 2½" x 1". It utilizes a MAN72 high brightness readout, and the MM5314 clock chip.

$17.95

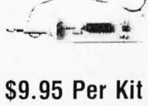


DIGITAL CLOCK KIT — 3½ INCH DIGITS
4 DIGIT KIT $49.95 **4 DIGIT ASSEMBLED $59.95**
6 DIGIT KIT $69.95 **6 DIGIT ASSEMBLED $79.95**

This clock features big 3½" high digits for viewing in offices, auditoriums, etc. Each digit is formed by 31 bright 0.2" LED's. The clock operates from 117 VAC, has either 12 or 24 hr. operation. The 6 digit version is 27" x 3½" x 1½" and the 4 digit is 18" x 3½" x 1½". Kits come complete with all components, case and transformer.

Specify 12 or 24 Hour When Ordering

JE803 PROBE
The Logic Probe is a unit which is for the most part indespensible in trouble shooting logic families TTL, DTL, RTL, CMOS. It derives the power it needs to operate directly off of the circuit under test, drawing a scant 10 mA max. It uses a MAN3 readout to indicate any of the following states by these symbols: (H) - 1 (LOW) - o (PULSE) - P. The Probe can detect high frequency pulses to 45 MHz. It can't be used at MOS levels or circuit damage will result.



$9.95 Per Kit
printed circuit board

T²L 5V 1A Supply



This is a standard TTL power supply using the well known LM309K regulator IC to provide a solid 1 AMP of current at 5 volts. We try to make things easy for you by providing everything you need in one package, including the hardware for only **$9.95 Per Kit**

7400N TTL

SN7400N	.16	SN7459A	.25		
SN7401N	.16	SN7460N	.22	SN74154N	1.00
SN7402N	.21	SN7470N	.45	SN74155N	.99
SN7403N	.16	SN7472N	.39	SN74156N	.99
SN7404N	.18	SN7473N	.37	SN74157N	.99
SN7405N	.24	SN7474N	.32	SN74160N	1.25
SN7406N	.20	SN7475N	.50	SN74161N	.99
SN7407N	.29	SN7476N	.32	SN74163N	.99
SN7408N	.25	SN7479N	5.00	SN74164N	1.10
SN7409N	.25	SN7480N	.50	SN74165N	1.10
SN7410N	.18	SN7482N	.98	SN74166N	1.25
SN7411N	.30	SN7483N	.70	SN74167N	5.50
SN7412N	.33	SN7485N	.89	SN74170N	2.10
SN7413N	.45	SN7486N	.39	SN74172N	8.95
SN7414N	.70	SN7488N	3.50	SN74173N	1.50
SN7416N	.35	SN7489N	2.49	SN74174N	1.25
SN7417N	.35	SN7490N	.45	SN74175N	.99
SN7420N	.21	SN7491N	.75	SN74176N	.90
SN7421N	.33	SN7492N	.49	SN74177N	.90
SN7422N	.49	SN7493N	.49	SN74180N	.99
SN7423N	.37	SN7494N	.79	SN74181N	2.49
SN7425N	.29	SN7495N	.79	SN74182N	.95
SN7426N	.29	SN7496N	.89	SN74184N	1.95
SN7427N	.37	SN7497N	4.00	SN74185N	2.20
SN7429N	.42	SN74100N	1.00	SN74186N	15.00
SN7430N	.26	SN74107N	.39	SN74187N	6.00
SN7432N	.31	SN74121N	.39	SN74188N	3.95
SN7437N	.27	SN74122N	.39	SN74190N	1.19
SN7438N	.27	SN74123N	.50	SN74191N	1.25
SN7439N	.25	SN74125N	.60	SN74192N	.89
SN7440N	.15	SN74126N	.60	SN74193N	.89
SN7441N	.89	SN74132N	1.09	SN74194N	1.25
SN7442N	.59	SN74136N	.95	SN74195N	.75
SN7443N	.75	SN74141N	1.15	SN74196N	1.25
SN7444N	.75	SN74142N	4.00	SN74197N	.75
SN7445N	.75	SN74143N	4.50	SN74198N	1.75
SN7446N	.81	SN74144N	4.50	SN74199N	1.75
SN7447N	.69	SN74145N	1.15	SN74200N	5.59
SN7448N	.79	SN74147N	2.35	SN74279N	.90
SN7450N	.26	SN74148N	2.00	SN74251N	1.79
SN7451N	.27	SN74150N	1.00	SN74284N	6.00
SN7453N	.27	SN74151N	.79	SN74285N	6.00
SN7454N	.20	SN74153N	.89	SN74367N	.75

MANY OTHERS AVAILABLE ON REQUEST
20% Discount for 100 Combined 7400's

CMOS

CD4000	.25			74C04N	.75
CD4001	.25			74C10N	.65
CD4002	.25	CD4035	1.85	74C20N	.65
CD4006	2.50	CD4040	2.45	74C30N	.65
CD4007	.25	CD4042	1.90	74C42N	2.15
CD4009	.59	CD4044	1.50	74C73N	1.50
CD4010	.59	CD4046	2.51	74C74	1.15
CD4011	.25	CD4047	2.75	74C90N	3.00
CD4012	.25	CD4049	.79	74C95N	2.00
CD4013	.47	CD4050	.79	74C107N	1.25
CD4016	.56	CD4051	2.95	74C151	2.90
CD4017	1.35	CD4053	2.95	74C154	4.00
CD4019	.55	CD4060	3.25	74C157	2.15
CD4020	1.49	CD4066	1.75	74C160	3.25
CD4022	1.25	CD4069	.45	74C161	3.25
CD4023	.25	CD4071	.45	74C163	3.00
CD4024	1.50	CD4081	.45	74C164	3.25
CD4025	.25	CD4511	2.50	74C173	2.60
CD4027	.69	CD4518	2.50	74C193	2.75
CD4028	1.65	MC14566	3.00	74C195	2.75
CD4029	2.90	74C00N	.39	MC4044	4.50
CD4030	.65	74C02N	.55	MC14016	.56

LINEAR

LM300H	.80			LM1351N	1.65
LM301H	.35			LM1414N	1.75
LM301CN	.35			LM1458C	.65
LM302H	.75	LM370N	1.1.5	LM1496N	.95
LM304H	1.00	LM373N	3.25	LM1556V	1.85
LM305H	.95	LM377N	4.00	LM2111N	1.95
LM307CN	.35	LM380N	1.39	LM2901N	2.95
LM308H	1.00	LM380CN	1.05	LM3065N	.69
LM308CN	1.00	LM381N	1.79	LM3900N	.55
LM309H	1.10	LM382N	1.79	LM3905N	.60
LM309K	.99	NE501K	8.00	LM3909	1.25
LM310CN	1.15	NE510A	6.00	LM5556N	1.85
LM311H	.90	NE531H	3.00	MC5558V	1.00
LM311N	.90	NE536T	6.00	LM7525N	.90
LM318CN	1.50	NE540L	6.00	LM7535N	1.25
LM319N	1.30	NE550N	.79	8038B	4.95
LM320K-5	1.35	NE555V	.39	LM75450	.49
LM320K-5.2	1.35	NE560B	5.00	75451CN	.39
LM320K-12	1.35	NE561B	5.00	75452CN	.39
LM320K-15	1.35	NE562B	5.00	75453CN	.39
LM320T-5	1.75	NE565H	1.25	75454CN	.39
LM320T-5.2	1.75	NE565N	1.75	75491CN	.79
LM320T-8	1.75	NE566CN	1.25	75492CN	.89
LM320T-12	1.75	NE567H	1.95	75494CN	.89
LM320T-15	1.75	NE567V	1.50	**RCA LINEAR**	
LM320T-18	1.75	LM703CN	.45	CA3013	2.15
LM320T-24	1.75	LM709H	.29	CA3023	2.56
LM323K-5	9.95	LM709N	.29	CA3035	2.48
LM324N	1.80	LM710N	.79	CA3039	1.35
LM339N	1.70	LM711N	.39	CA3046	1.30
LM340K-5	1.95	LM723H	.55	CA3059	3.25
LM340K-6	1.95	LM723H	.55	CA3060	3.25
LM340K-8	1.95	LM733N	1.00	CA3080	.85
LM340K-12	1.95	LM739N	1.00	CA3081	2.00
LM340K-15	1.95	LM741CH	.35	CA3082	2.00
LM340K-18	1.95	LM741CN	.35	CA3083	1.60
LM340K-24	1.95	LM741-14N	.39	CA3086	.85
LM340T-5	1.75	LM747H	.79	CA3089	3.75
LM340T-6	1.75	LM747N	.79	CA3091	10.20
LM340T-8	1.75	LM748H	.39	CA3102	2.95
LM340T-12	1.75	LM748N	.39	CA3123	2.15
LM340T-15	1.75	LM1303N	.90	CA3130	1.39
LM340T-18	1.75	LM1304N	1.19	CA3140	1.25
LM340T-24	1.75	LM1305N	1.40	CA3600	1.75
LM350N	1.00	LM1307N	.85	RC4194	5.95
LM351CN	.65	LM1310N	2.95	RC4195	3.25

74LS00 TTL

74LS00	29			74LS139	1.95
74LS02	29			74LS151	1 55
74LS03	29	74LS74	49	74LS153	1 89
74LS04	35	74LS75	69	74LS157	1 55
74LS05	35	74LS76	49	74LS162	2 25
74LS08	29	74LS83	1 75	74LS163	2 25
74LS10	29	74LS85	2 49	74LS164	1 95
74LS13	69	74LS86	49	74LS175	1.95
74LS14	1 75	74LS90	1 25	74LS181	3 69
74LS20	29	74LS92	1 25	74LS190	2 85
74LS26	39	74LS93	1 25	74LS191	2 85
74LS27	39	74LS95	1 95	74LS192	2 85
74LS28	39	74LS96	1 89	74LS193	2 85
74LS30	29	74LS107	59	74LS194	1 89
74LS32	39	74LS109	59	74LS195	1 89
74LS40	39	74LS112	59	74LS257	1 75
74LS51	29	74LS132	1 25	74LS260	55
74LS55	29	74LS136	59	74LS279	79
74LS73	49	74LS138	1 89	74LS670	3 95

CLOCK CHIPS

MM5309	6 Digit, BCD Outputs, Reset PIN.	$9.95
MM5311	6 Digit, BCD Outputs, 12 or 24 Hour	4.95
MM5312	4 Digit, BCD Outputs, 1 PPS Output	4.95
MM5314	6 Digit, 12 or 24 Hour, 50 or 60 Hz	4.95
MM5316	4 Digit, Alarm, 1 PPS Output	6.95
MM5318	Video Clock Chip, For Use With (MM5841 - $9.95)	9.95
CT7001	6 Digit, Calendar, Alarm, 12 or 24 Hour	5.95

DATA HANDBOOKS

7400	Pin-out & Description of 5400/7400 ICS	$2.95
CMOS	Pin-out & Description of 4000 Series ICS	$2.95
Linear	Pin-out & Functional Description	$2.95

ALL THREE HANDBOOKS $6.95

There is only *one*

James ELECTRONICS

They can try and copy our ad,
BUT
They can't copy our service!

DISCRETE LEDS

.125" dia.			.190" dia.		
XC209	Red	10/$1	XC111	Red	10/$1
XC209	Green	4/$1	XC111	Green	4/$1
XC209	Orange	4/$1	XC111	Yellow	4/$1
XC209	Yellow	4/$1	XC111	Orange	4/$1

.200" dia			.185" dia.			.200" dia.			.085" dia.
XC22	Red	10/$1	XC526	Red	10/$1	XC556	Red	10/$1	MV50 - Red - 6/$1
XC22	Green	4/$1	XC526	Green	4/$1	XC556	Green	4/$1	INFRA-RED LED
XC22	Yellow	4/$1	XC526	Yellow	4/$1	XC556	Yellow	4/$1	¼" x ¼" x 1/16"
XC22	Orange	4/$1	XC526	Orange	4/$1	XC556	Orange	4/$1	Flat 5/$1.00
SSL-22	RT	4/$1	XC526	Clear	4/$1	XC556	Clear	7/$1	

SPECIAL * — XC556 Red 100/$8.00 1000/$60.00 — SPECIAL *

DISPLAY LEDS

DL707 DL33B

TYPE	POLARITY	HT		TYPE	POLARITY	HT	
MAN 1	Common Anode	270	2.95	MAN 3640	Common Cathode-orange	.300	1.75
MAN 2	5 x 7 Dot Matrix	.300	4.95	MAN 4710	Common Anode-Red	.400	1.95
MAN 3	Common Cathode	125	3/1.00	DL701	Common Anode-red ±	.300	99
MAN 4	Common Cathode	187	1.95	DL704	Common Cathode	.300	99
MAN 7	Common Anode	.300	1.25	DL707	Common Anode	.300	99
MAN 7G	Common Anode-green	.300	1.95	MAN 4740	Common Anode-Red	.400	.99
MAN 7Y	Common Anode-yellow	.300	1.95	DL741	Common Anode	.600	1.50
MAN 52	Common Anode-green	.300	.99	DL 747	Common Anode	600	2.25
MAN 64	Common Anode-red	.400	.99	DL 750	Common Cathode	600	2.49
MAN 74	Common Cathode	.300	1.50	DL 33B	Common Cathode	110	.50
MAN 82	Common Anode-yellow	.300	.99	FND70	Common Cathode	.250	.75
MAN 84	Common Cathode-yellow	.300	.99	FND503	Common Cathode	.500	1.00
MAN 3620	Common Anode -orange	.300	1.75	FND507	Common Anode	.500	1.00

FACTORY REJECTS ATARI GAME BOARDS

BOARD A — 8½" x 16" — **$6.95 ea.** ONLY 500 EA. AVAILABLE
Over 60 each reusable IC's
Misc. Transistors, Resistors, Diodes, Caps, Crystals, Switch, etc.

DELUXE BOARD B — 11½" x 18" — **$9.95 ea.** ONLY 500 EA. AVAILABLE
Over 100 each reusable IC's
Misc. Tranistors, Resistors, Diodes, Caps, Crystals, Switches, LEDS, etc.

NEW AY-3-8500-1 TV GAME CHIP

- Score display
- Selectable angles, speed and bat size
- Auto or manual ball service
- Realism sounds
- 6 Games — tennis, hockey/soccer, squash, pelota/practice and two rifle games

28 Lead Dual In Line

The circuit is intended to be battery powered. Minimum number of external components required for complete system.

$24.95

IC SOLDERTAIL — LOW PROFILE (TIN) SOCKETS

	1-24	25-49	50-100		1-24	25-49	50-100
8 pin	$.17	.16	.15	24 pin	$.38	.37	.36
14 pin	.20	.19	.18	28 pin	.45	.44	.43
16 pin	.22	.21	.20	36 pin	.60	.59	.58
18 pin	.29	.28	.27	40 pin	.63	.62	.61
22 pin	.37	.36	.35				

SOLDERTAIL STANDARD (TIN)

14 pin	$.27	.25	.24	28 pin	$.99	.90	.81
16 pin	.30	.27	.25	36 pin	1.39	1.26	1.15
18 pin	.35	.32	.30	40 pin	1.59	1.45	1.30
24 pin	.49	.45	.42				

SOLDERTAIL STANDARD (GOLD)

8 pin	$.30	.27	.24	24 pin	$.70	.63	.57
14 pin	.35	.32	.29	28 pin	1.10	1.00	.90
16 pin	.38	.35	.32	36 pin	1.75	1.40	1.26
18 pin	.52	.47	.43	40 pin	1.75	1.59	1.45

WIRE WRAP SOCKETS (GOLD) LEVEL #3

10 pin	$.45	.41	.37	24 pin	$1.05	.95	.85
14 pin	.39	.38	.37	28 pin	1.40	1.25	1.10
16 pin	.43	.42	.41	36 pin	1.59	1.45	1.30
18 pin	.75	.68	.62	40 pin	1.75	1.55	1.40



Plastic Push Button Switch

- 18 AWG Solid Wire - 5" Long
- .50 (wide) X .60 (high) ⅜-27 Thread
- 8 AMP @ 14 Volt - 1 AMP @ 110 Volt

		1-9	10-Up
J-188-1	Push On-Push Off	.59	.49
J-188-2	Normally Open	.59	.49
J-188-3	Normally Closed	.59	.49

MINATURE TOGGLE SWITCH



JMT-221	DPDT	on/off/on	$1.95
JMT-223	DPDT	on/none/on	$1.75
JMT-121	SPDT	on/off/on	$1.50
JMT-123	SPDT	on/none/on	$1.25



CLIPLITE 8/$1.49

LED MOUNTING SYSTEM
use with XC556 LEDS
Specify Colors — Red - Green - Amber - Yellow



DIP SWITCHES

SPST Slide Action

#206-4 (8 pin dip) 4 switch unit	$1.75 ea.
#206-7 (14 pin dip) 7 switch unit	$1.95 ea.
#206-8 (16 pin dip) 8 switch unit	$2.25 ea.

50 PCS. RESISTOR ASSORTMENTS $1.75 PER ASST.

ASST. 1	5 ea.	10 OHM 27 OHM	12 OHM 33 OHM	15 OHM 39 OHM	18 OHM 47 OHM	22 OHM 56 OHM	1/4 WATT 5% 50 PCS.
ASST. 2	5 ea.	68 OHM 180 OHM	82 OHM 220 OHM	100 OHM 270 OHM	120 OHM 330 OHM	150 OHM 390 OHM	1/4 WATT 5% 50 PCS.
ASST. 3	5 ea.	470 OHM 1.2K	560 OHM 1.5K	680 OHM 1.8K	820 OHM 2.2K	1K 2.7K	1/4 WATT 5% 50 PCS.
ASST. 4	5 ea.	3.3K 8.2K	3.9K 10K	4.7K 12K	5.6K 15K	6.8K 18K	1/4 WATT 5% 50 PCS.
ASST. 5	5 ea.	22K 56K	27K 68K	33K 82K	39K 100K	47K 120K	1/4 WATT 5% 50 PCS.
ASST. 6	5 ea.	150K 390K	180K 470K	220K 560K	270K 680K	330K 820K	1/4 WATT 5% 50 PCS.
ASST. 7	5 ea.	1M 2.7M	1.2M 3.3M	1.5M 3.9M	1.8M 4.7M	2.2M 5.6M	1/4 WATT 5% 50 PCS.

ASST. 8R Includes Resistor Assortments 1-7 (350 PCS.) **$10.95 ea.**

$5.00 Minimum Order — U.S. Funds Only
California Residents — Add 6% Sales Tax

Spec Sheets - 25¢ — Send 35¢ Stamp for 1977A Catalog
Dealer Discount Available — Request Pricing

NOW OPEN SATURDAYS

James ELECTRONICS

1021-A HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097
All Advertised Prices Good Thru May

WIRE WRAP CENTER

HOBBY-WRAP TOOL-BW-630



- Battery Operated (Size C)
- Weighs ONLY 11 Ounces
- Wraps 30 AWG Wire onto Standard DIP Sockets (.025 inch)
- Complete with built-in bit and sleeve

$34.95
(batteries not included)

WIRE-WRAP KIT — WK-2-W



WRAP • STRIP • UNWRAP

- Tool for 30 AWG Wire
- Roll of 50 Ft. White or Blue 30 AWG Wire
- 50 pcs. each 1", 2", 3" & 4" lengths — pre-stripped wire.

$11.95



WIRE WRAP TOOL WSU-30

WRAP • STRIP • UNWRAP - $5.95

WIRE WRAP WIRE — 30 AWG

25 ft. min. $1.25 50 ft. $1.95 100 ft. $2.95 1000 ft. $15.00
SPECIFY COLOR — White - Yellow - Red - Green - Blue - Black

CUTTER CRIMPER TOOL (CS-8)



- Plier Nose (serrated-jaw)
- Scissors Action Cutting
- 6 Bolt Cutters (4-40, 5-40, 6-32, 8-32, 10-32, 10-24)
- Crimp Stations (7mm Auto — 22-20 to 12-X0 elect.)
- "Up-Front" Wire Cutting
- Scissors Action Stripping (No. 22-20 to No. 10)
- Crimp Stations — insulated (2w-20 to 12-10 elect.)

Actual Size - 8¼" length **$8.50**



Permacel Electrical Tape

• ¾" (wide) X 66 ft. (long) • All weather • Not import

$1.25 per roll — $9.95 per 10 roll package

ZENERS — DIODES — RECTIFIERS

TYPE	VOLTS	W	PRICE	TYPE	VOLTS	W	PRICE
1N746	3.3	400mm	4/1.00	1N4005	600 PIV	1 AMP	10/1.00
1N751A	5.1	400m	4/1.00	1N4006	800 PIV	1 AMP	10/1.00
1N752	5.6	400m	4/1.00	1N4007	1000 PIV	1 AMP	10/1.00
1N753	6.2	400m	4/1.00	1N3600	50	200m	6/1.00
1N754	6.8	400m	4/1.00	1N4148	75	10m	15/1.00
1N959	8.2	400m	8/1.00	1N4154	35	10m	12/1.00
1N965B	15	400m	4/1.00	1N4305	75	25m	20/1.00
1N5232	5.6	500m	28	1N4734	5.6	1w	28
1N5234	6.2	500m	28	1N4735	6.2	1w	28
1N5235	6.8	500m	28	1N4736	6.8	1w	28
1N5236	7.5	500m	28	1N4738	8.2	1w	28
1N456	25	40m	6/1.00	1N4742	12	1w	28
1N458	150	7m	6/1.00	1N4744	15	1w	28
1N485A	180	10m	6/1.00	1N1183	50 PIV	35 AMP	1.60
1N4001	50 PIV	1 AMP	12/1.00	1N1184	100 PIV	35 AMP	1.70
1N4002	100 PIV	1 AMP	12/1.00	1N1185	150 PIV	35 AMP	1.50
1N4003	200 PIV	1 AMP	12/1.00	1N1186	200 PIV	35 AMP	1.80
1N4004	400 PIV	1 AMP	12/1.00	1N1188	400 PIV	35 AMP	3.00

SCR AND FW BRIDGE RECTIFIERS

C36D	15A @ 400V	SCR	$1.95
C38M	35A @ 200V	SCR	1.95
2N2328	1.6A @ 200V	SCR	.50
MDA 980-1	12A @ 50V	FW BRIDGE REC.	1.95
MDA 980-3	12A @ 200V	FW BRIDGE REC.	1.95

TRANSISTORS

MPS A05	5/$1.00			PN4249	4/$1.00
MPS A06	5/$1.00			PN4250	4/$1.00
2N2219A	3/$1.00	PN3567	3/$1.00	2N4400	4/$1.00
2N2221	4/$1.00	PN3568	4/$1.00	2N4401	4/$1.00
2N2222A	5/$1.00	PN3569	4/$1.00	2N4402	4/$1.00
2N2369	5/$1.00	2N3704	5/$1.00	2N4403	4/$1.00
2N2369A	4/$1.00	2N3705	5/$1.00	2N4409	5/$1.00
2N2484	4/$1.00	2N3706	5/$1.00	2N5086	4/$1.00
2N2906A	4/$1.00	2N3707	5/$1.00	2N5087	4/$1.00
2N2907A	5/$1.00	2N3711	5/$1.00	2N5088	4/$1.00
2N2925	5/$1.00	2N3724	$.65	2N5089	4/$1.00
2N3053	2/$1.00	2N3725	$1.00	2N5129	5/$1.00
2N3055	$.89	2N3903	5/$1.00	2N5138	5/$1.00
MJE3055	$1.00	2N3904	4/$1.00	2N5139	5/$1.00
MJE2955	$1.25	2N3905	4/$1.00	2N5209	5/$1.00
2N3392	5/$1.00	2N3906	4/$1.00	2N5951	5/$1.00
2N3398	5/$1.00	2N4013	3/$1.00	C106B1SCR	2/$1.00
		2N4014	3/$1.00	2N5432	$2.00
		2N4123	10/$1.00		

CAPACITOR CORNER

50 VOLT CERAMIC DISC CAPACITORS

	1-9	10-49	50-100		1-9	10-49	50-100
10 pf	.05	.04	.03	.001µF	.05	.04	.035
22 pf	.05	.04	.03	.0047µF	.05	.04	.035
47 pf	.05	.04	.03	.01µF	.05	.04	.035
100 pf	.05	.04	.03	.022µF	.06	.05	.04
220 pf	.05	.04	.03	.047µF	.06	.05	.04
470 pf	.05	.04	.035	.1µF	.12	.09	.075
100 VOLT MYLAR FILM CAPACITORS							
.001mf	.12	.10	.07	.022mf	.13	.11	.08
.0022	.12	.10	.07	.047mf	.21	.17	.13
.0047mf	.12	.10	.07	.1mf	.27	.23	.17
.01mf	.12	.10	.07	.22mf	.33	.27	.22
+20% DIPPED TANTALUMS (SOLID) CAPACITORS							
.1/35V	.28	.23	.17	1.5/35V	.30	.26	.21
.15/35V	.28	.23	.17	2.2/25V	.31	.27	.22
.22/35V	.28	.23	.17	3.3/25V	.31	.27	.22
.33/35V	.28	.23	.17	4.7/25V	.32	.28	.23
.47/35V	.28	.23	.17	6.8/25V	.36	.31	.25
.68/35V	.28	.23	.17	10/25V	.40	.35	.29
1.0/35V	.28	.23	.17	15/25V	.63	.50	.40
MINIATURE ALUMINUM ELECTROLYTIC CAPACITORS							
Axial Lead				**Radial Lead**			
.47/50V	.15	.13	.10	.47/25V	.15	.13	.10
1.0/50V	.16	.14	.11	.47/50V	.16	.14	.11
3.3/50V	.15	.13	.10	1.0/16V	.15	.13	.10
4.7/25V	.16	.14	.12	1.0/25V	.16	.14	.11
10/25V	.15	.13	.10	1.0/50V	.16	.14	.11
10/50V	.16	.14	.12	4.7/16V	.15	.13	.10
22/25V	.17	.15	.12	4.7/25V	.15	.13	.10
22/50V	.24	.20	.18	4.7/50V	.16	.14	.11
47/25V	.19	.17	.15	10/16V	.14	.12	.09
47/50V	.25	.21	.19	10/25V	.15	.13	.10
100/25V	.24	.20	.18	10/50V	.16	.14	.12
100/50V	.35	.30	.28	47/50V	.24	.21	.19
220/25V	.32	.28	.25	100/16V	.19	.15	.14
220/50V	.45	.41	.38	100/25V	.24	.20	.18
470/25V	.33	.29	.27	100/50V	.35	.30	.28
1000/16V	.55	.50	.45	220/16V	.23	.17	.16
2200/16V	.70	.62	.55	470/25V	.31	.28	.26

computer display terminal

This display terminal has an integral controller, B/W cathode ray tube and keyboard. The system has a serial I/O interface for communication and an I/O interface for a printer



External logic & power pack not shown.

DISPLAY (P/N 4802-1095-501) FEATURES:

- 17" B/W CRT
- 41 lines of data
- 52 characters per line
- Characters are generated by a diode matrix "graphic" technique
- 21 special push-buttons wired for a program call up
- Brightness Control
- Self-contained power supply

KEYBOARD (P/N 4802-1115-501) FEATURES:

- Reed switch technology
- 54 data keys
- 28 special keys detachable with cable

LOGIC UNIT (P/N 4802-1157-502) FEATURES:

- 1024 by 6 bit core memory
- Printer I/O interface
- Communication I/O interface

POWER: 115V, 50/60 Hz, 500 Watts

WEIGHT: 210 lbs. (including logic unit, keyboard, display and cables.)

FOB LYNN MASS (you pay shipping)
Check with order please.

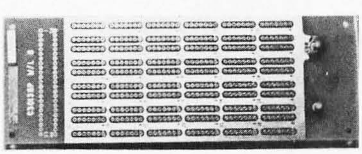
$180.00

"AS IS"

4 way cursor control, graphics display.

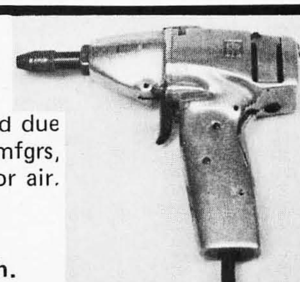
The story: These are unused terminals made for airport ticketing & seat assignment. After several years of storage they require tinkering to make operable. We have some hints printed such as cleaning PC fingers. One of our customers has this tied into his KIM-1, another has his running with his IMSAI. We have data on this. Should be useable on most common computers. A hell of a deal and all for a paltry $180.00. Don't be left out as many were on our past VIATRON deal. Sold "as is" all sales final.

WITH COMPLETE DOCUMENTATION



Here is a real deal in a PC module 6x5 sockets (30). List price over $50 each, most by AUGAT. Some pre-wired. New, unused boxed, 14 or 16 pin, 5 x 6 sockets. $15.00 each or 2 for $25, state your choice 14 or 16 pin.

WIRE WRAP GUNS

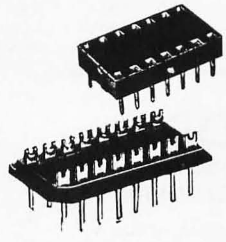


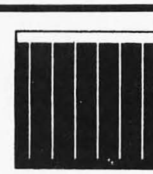
Used wire wrap guns, released due to factory closure. Various mfgrs, some Ingersol Rand, electric or air. No collets. State choice.
Cost over $100.00 each

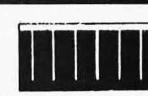
Our price only $15.00 each.

IC Sockets, while they last . . .

8 Pin	10/$1.00
14 Pin	10/$1.25
16 Pin	10/$1.50
18 Pin	10/$1.75
14 Pin IC connector	10/$1.25







shown actual size

SOLAR CELLS

Designed for the space program, these are the highly efficient silicon high output cells. Used for powering equipment, charging batteries. Made by Ion Physics Corp. Each with spec sheet.
Size .394 x .788" 65 mA, .43 V $1.25 12/$12.00
Size .788 x .788" 125 mA, .43 V $1.60 12/$15.00

Meshna

Please add shipping cost on above. Minimum order $10
FREE CATALOG SP-9 NOW READY
P.O. Box 62, E. Lynn, Massachusetts 01904

ADD CALCULATOR FUNCTIONS without adding more memory to your micro. The R.G.S. mathmatical function unit uses a calculator chip to provide floating point add, sub, mult, divide, log, trig & power functions for your micro.
PC BOARD AND MANUAL.......................................$27.00

TOP QUALITY $28

HEAVY DUTY POWER TRANSFORMER FOR IMSAI/ALTAIR



5VDC @ 22A
#18 VDC 40V (Prom programming)

INCLUDE APPROX SHIPPING 20 LBS
LIMITED QUANTITY

Check or money order only. Calif. residents 6% tax. All orders postpaid in the U.S. $10 Min. order. Prices subject to change without notice.

the proko electronics shoppe

439 marsh st.
san luis obispo, ca. 93401
805/544-5441



THE PROKO PAPER TAPE READER: A fully TTL buffered optical tape reader for reading 8 level paper tape. It's compatible with nearly all parallel input ports and has both positive & negative strobe outputs. The power (+5V @ 20ma) is derived from your I/O board... Just supply a light source (study lamp), grab the tape and pull!
PTR - II KIT - $54.00 ASSEMBLED $68.

* JUST IN:

★ 2102AL-4 - FULL SPEED MEMORY (500ns) ★

★ Nearly 1/3 less power consumption of even a 21L02A ★

2.00 EACH — 32 FOR 1.80 EACH
★ 64 - 1.70 — 128 FOR 1.60 EACH ★

82S06	2.00	82S126	3.50	74C200	5.50
82S07	2.00	82S129	3.50	8573	4.50
82S11	2.00	82S130	3.95	8574	5.50
82S12	2.00	82S131	3.95	8575	4.50
82S17	2.00	74S206	2.10	8576	4.50
82S23	2.50	74S412	4.00	8577	3.50
82S123	3.00			8578	4.00

7400	.16	7473	.35	74164	1.10
7401	.16	7474	.35	74165	1.10
7402	.21	7475	.50	74166	1.25
7403	.16	7476	.30	74170	2.10
7404	.18	7480	.50	74173	1.50
7405	.24	7483	.70	74174	1.95
7406	.20	7485	.90	74175	.95
7407	.29	7486	.40	74176	.90
7408	.25	7489	2.00	74177	.90
7409	.25	7490	.45	74179	.90
7410	.18	7491	.75	74180	.90
7411	.30	7492	.50	74181	2.50
7413	.45	7493	.50	74182	.95
7414	.70	7494	.80	74184	1.95
7416	.35	7495	.75	74185	2.20
7417	.35	7496	.90	74190	1.15
7420	.20	74100	1.00	74191	1.25
7423	.37	74107	.40	74192	.90
7425	.30	74109	.90	74193	.90
7426	.30	74121	.40	74194	1.25
7427	.35	74122	.50	74195	.75
7430	.25	74123	.70	74196	1.25
7432	.30	74125	.60	74197	.90
7437	.27	74126	.60	74198	1.75
7438	.27	74132	1.00	74199	1.75
7440	.15	74141	1.15	74200	4.95
7441	.85	74145	1.15	74251	1.75
7442	.60	74147	2.35	74284	4.95
7443	.75	74148	2.00	74285	4.95
7444	.75	74150	1.00	74365	.90
7445	.75	74151	.80	74367	.75
7446	.80	74153	.90	74368	.90
7447	.70	74154	1.00	MH0025	2.50
7448	.80	74155	1.00	MH0026	2.95
7450	.25	74156	1.00	95H90	9.95
7451	.25	74157	1.00		
7453	.25	74160	1.25		
7454	.20	74161	1.00		
7460	.20	74162	1.50	1488	1.50
7470	.45	74163	1.00	1489	1.50
7472	.40				

90 Day Guarantee on SSM Products. Kits MB-2, MB-3 (2K OR 4K), MB-4, MB-6, 10-2 video board and mother board with connectors may be combined for a discount of 10% in quantities of 10 or more. This supercedes the flier of 13 Sept. 1976.

KITS BY cybercom

A DIVISION OF SOLID STATE MUSIC

4Kx8 Static Memories
MB-1 MK-8 board, 1 usec 2102 or eq. PC Board..... $22 (NOT FOR S-100 BUSS)

MB-2 Altair 8800 or IMSAI compatible Switched address and wait cycles. PC Board......$25

MB-4 Improved MB-2 designed for 8K "piggy-back" without cutting traces. PC Board.................. $30
Kit 4K .5 usec.....$129.95 Kit 8K .5 usec.....$199

MB-3 1702A's Eroms, Altair 8800 & Imsai 8080 compatible Switched address & wait cycles. 2K may be expanded to 4K. Kit less Proms.....$65 2K Kit.....$145 4K Kit......$225

MB-6 PC Bd...$35

MB-6A 8Kx8 Switched address and wait assignments. Memory protection is switchable for 256, 512, 1K, 2K, 4K and 8K. 91L02A .5 usec rams, Altair 8800 & IMSAI compatible. With battery power option
Kit.......$250 Assembled & tested......$290

I/O Boards
I/0-2 I/0 for 8800, 2 ports, committed pads for 3 more, other pads for EROMS UART, etc.
Kit............$47.50 PC Board only..........$25

64 x 16 VIDEO BOARD Altair plug compatible display 32 x 16 or 64 x 16 switch selectable. Composite and parallel video ports, upper and lower case with software.
Kit.. $179.95
PC board...$35

SP-1 Synthesizer board. Computer controled wave forms 9 octaves 1 vrms ½% distortion includes high level music language.
KIT...$250.

Altair compatable mother board. Room for 15 connectors 11" x 11½" (w/o connectors).......$45
With 15 connectors......$110.00

Altair extender board (w/o connectors)......$9.50
With w/w connector.......$13.50
100 Pin spec WW or Soldertail both fit IMSAI or SSM Mother Bd $5.00 ea 10/$44

1702A* EROM	$10.00
*programming send hex list	5.00
AY5-1013 UART	$6.95
2513 prime spec. upper or lower case	11.00
8080A prime CPU	25.00
8212 prime latch buffer	4.00
8224 prime clock gen	5.00
8228 prime sys controller	8.90

MM5309	8.00	2501B	$1.25	1101	1.25	74LS00	.40	74L00	.25	74L78	.90
MM5312	4.00	2503V	2.00	1103	1.25	74LS01	.50	74L01	.25	74L85	1.40
MM5313	4.00	2504V	2.00	2101	4.50	74LS02	.40	74L02	.25	74L86	.75
MM5320	4.95	2505K	2.00	2111-1	4.50	74LS03	.40	74L03	.25	74L89	3.50
MM5554	1.90	2507V	1.25	2112	4.50	74LS04	.45	74L04	.30	74L90	1.50
MM5556	2.50	2509A	2.00	2602	1.60	74LS05	.45	74L05	.40	74L91	1.50
MM5055	1.90	2510A	2.00	4002-1	7.50	74LS10	.40	74L06	.30	74L93	1.70
DM8836	.60	2511A	2.80	4002-2	7.50	74LS11	.50	74L08	.40	74L95	1.70
DM8837	1.50	2517V	1.25	MM5262	1.00	74LS12	.55	74L09	.40	74L98	2.80
80C95	1.10	2518B	1.50	7489	2.00	74LS20	.40	74L10	.30	74L123	1.50
80C97	1.00	2519B	2.80	74200	4.95	74LS22	.45	74L20	.35	74L154	2.00
80L97	1.50	2521V	1.50	74S89	3.50	74LS27	.45	74L26	.40	74L164	2.50
81L22	1.50	2522V	2.00	74C89	3.50	74LS30	.40	74L30	.40	74L165	2.50
81L23	1.90	2525V	2.80	74L89	3.50	74LS42	1.50	74L32	.45	74L192	1.25
85L51	2.50	2527V	2.80	8223	2.50	74LS55	.40	74L42	1.50	74L193	1.20
85L52	2.50	2528V	2.80	8T80	2.50	74LS73	.65	74L51	.35	MC4044	2.25
85L63	1.25	2529V	2.80	8T97	2.00	74LS74	.65	74L54	.45	N8263	3.50
86L70	1.50	2532B	2.80	INTEL		74LS76	.65	74L55	.35	N8826	2.50
86L75	1.90	2533V	2.80	8216	4.95	74LS151	1.55	74L71	.30	DM8131	2.50
86L99	3.50	91L02APC	2.55	8214	8.30	74LS174	2.20	74L73	.55	8T16	2.00
86L12	.80	32 each	2.40	8251	14.50	74LS175	1.95	74L74	.55	8T20	2.00
8T13	2.50	64 each	2.25	8255	14.50	74LS192	2.85	74L75	1.20	8T10	2.00

MIKOS

419 Portofino Dr.
San Carlos, Ca. 94070

Please send for xistor, IC & kit list

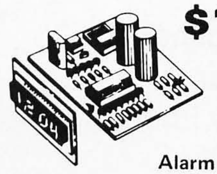
For large orders please send money order or cashiers check to avoid delays in waiting for checks to clear.

Check or money order only. Calif. resident 6% tax. All orders postpaid in U.S. All devices tested prior to sale. Money back 30 day guarantee. Sorry we can not accept returned IC's that have been soldered to. $10 min. order. Prices subject to change without notice.

Thinly disguised affiliates of KO Electronics and Surplus, S.L.O., CA 93401

S.D. SALES CO. P.O. BOX 28810 – B DALLAS, TEXAS 75228

JUMBO LED CAR CLOCK



$16.95 KIT

Alarm Option – $1.50
AC XFMR – $1.50

THE HOTTEST SELLING KIT WE EVER PRODUCED!

You requested it! Our first D.C. operated clock kit. Professionally engineered from scratch. Not a makeshift kluge as sold by others. Features:

A. Bowmar Jumbo –.5 inch LED array.
B. MOSTEK – 50250 – Super Clock Chip.
C. On board precision crystal time base.
D. 12 or 24 Hr. Real Time Format.
E. Perfect for cars, boats, vans, etc.
F. P.C. Board and all parts (less case) included.

50,000 SATISFIED CLOCK KIT CUSTOMERS CANNOT BE WRONG!

THIS MONTH'S SPECIALS

AMD – 8080A	$14.95
Z-80 CPU	49.95
82S129 1K PROM	2.50

1702A 2K EPROM

We tell it like it is. We could have said these were factory new, but here is the straight scoop. We bought a load of new computer gear that contained a quantity of 1702 A's in sockets. We carefully removed the parts, verified their quality, and are offering them on one heck of a deal. First come, first served. Satisfaction guaranteed! U.V. Eraseable. **$6.95 ea. 4/$25**

UP YOUR COMPUTER! 21L02-1 1K LOW POWER 500 NS STATIC RAM Time is of the essence!

And so is power. Not only are our RAM's faster than a speeding bullet but they are now very low power. We are pleased to offer prime new 21L02–1 low power and super fast RAM's. Allows you to STRETCH your power supply farther and at the same time keep the wait light off.

8 for $12.95

60 HZ CRYSTAL TIME BASE S.D. SALES EXCLUSIVE!

$5.95 ea. 2/$10.00

KIT FEATURES:

A. 60HZ output with accuracy comparable to a digital watch.
B. Directly interfaces with all MOS clock chips.
C. Super low power consumption (1.5 MA typ.)
D. Uses latest MOS 17 stage divider IC.
E. Eliminates forever the problem of AC line glitches.
F. Perfect for cars, boats, campers, or even for portable clocks at ham field days.
G. Small size; can be used in existing enclosures.

Kit includes Crystal, Driver IC, PC board, plus all necessary parts and specs. At last count – over 20,000 sold!

S.D. SALES EXCLUSIVE

$12.95 MOS 6 DIGIT UP–DOWN COUNTER $12.95

40 PIN DIP. Everything you ever wanted in a counter chip. Features: Direct LED segment drive, single power supply (12 VDC TYPE.), six decades up/down, pre-loadable counter, separate pre-loadable compare register with compare output, BCD and seven segment outputs, internal scan oscillator, CMOS compatible, leading zero blanking. 1MHZ. count input frequency. Very limited quantity! WITH DATA SHEET

7400—19c	7411—29c	7451—19c	7490—65c	74153—75c
74LS00—49c	7413—50c	7453—19c	74LS90—95c	74154-1.00
7402—19c	7416—69c	7473—39c	7492—75c	74157—75c
74LS02—49c	7420—19c	7474—35c	7493—69c	74161—95c
7404—19c	7430—19c	74LS74-59c	7495—75c	74164-1.10
74L04—29c	7432—34c	7475—69c	7496—89c	74165-1.10
74S04—44c	7437—39c	7476—35c	74121-38c	74174—95c
74LS04-49c	7438—39c	7480—49c	74123-65c	74181-2.50
7406—29c	7440—19c	7483—95c	74132-1.70	74191-1.25
7408—19c	7447—85c	7485—95c	74S138-1.95	74192-1.25
7410—19c	7448—85c	7486—45c	74141-75c	74193-1.00
				74195—69c

TTL INTEGRATED CIRCUITS

1000 MFD Filter Caps



Rated 35 WVDC Upright style with PC leads. Most popular value for hobbyists. Compare at up to $1.19 ea. from franchise type electronic parts stores. S.D. Special 4/$1.

Slide Switch Assortment



Our best seller. Includes miniature and standard sizes; single and multi-position units. All new, first name brand. Try one package and you'll reorder more! Special **12/$1.00**

RESISTOR ASSORTMENT

1/4W 5% & 10% PC leads. A good mix of values. **200/$2.**



P.C. LEAD DIODES

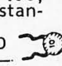
1N4148/1N914 100/$2.00
1N4002-1A. 100 PIV 40/$1.

HEAVY DUTY Full Wave Bridge

25 AMP 50 PIV
$1.25

Disc Cap Assortment

PC leads. At least 10 different values. Includes .001, .01, .05, plus other standard values. 60/$1.00



$9.95 KIT

P.C. Board – 3.00
AC XFMR – 1.50

Do not confuse with Non-Alarm kits sold by our competition! Eliminate the hassle – avoid the 5314!

SIX DIGIT ALARM CLOCK KIT

We made a fantastic kit even better. Redesigned to take advantage of the latest advances in I.C. clock technology. Features: Litronix Dual ½" displays, Mostek 50250 super clock chip, single I.C. segment driver, SCR digit drivers. Greatly simplified construction. More reliable and easier to build. Kit includes all necessary parts (except case). P.C.B. or XFMR optional. *NEW! WITH JUMBO LED READOUTS!*

Motorola SCR

2N4443. 8 AMP 400 PIV. P.C. Leads 3/$1.

FAIRCHILD - TBA 641

4W. Audio power Amp. Just out! In special heat sink DIP. One super audio IC. **$1.50 with data**

FND-359 -Led Readout

.4 IN. Common Cathode. High effeciency. Has FND-70 PIN OUT. **59c**

COMPUTER POWER SUPPLY

$15.95



A very fortunate purchase. One of the best industrial quality REGULATED supplies we have seen. High performance, small size. Input is 120 VAC 60 HZ. Has the following regulated outputs: **–5VDC@800MA; –15VDC @ 1.25 AMP; –25VDC @ 180 MA.** Sold at a fraction of original cost. Do yourself a favor and order NOW. We expect a quick sellout.

NEXT MONTH:
S.D. will have music for your ears. Watch our ads.

For your Imsai or Altair 8080 Computer:
Z–80 CPU Kit – $149. 4K Low Power Ram Kit – $89.95

OUR CATALOG

is chocked full of rare parts bargains, deals, RAM or CPU kits, plus much more. Yours FREE!

PRICES SHOWN SUBJECT TO CHANGE WITHOUT NOTICE.

Terms: Money back guarantee. No COD. Texas residents add 5% sales tax. Add 5% of order for postage & handling. Orders under $10. add 75c. Foreign orders: US funds only!

Call your Bankamericard or Master Charge order in on our continental United States toll free Watts:

1–800–527-3460
Texas Residents Call Collect:
214/271-0022

Special Thanks to:
Dennis, Fred, Abe, Bill, Sam, Hal, Tom, Alex, John, Ely, and Larry

S.D. SALES CO.
P. O. BOX 28810 B
Dallas, Texas 75228

ORDERS OVER $15.00 – CHOOSE $1.00 FREE MERCHANDISE

F8 EVALUATION BOARD KIT WITH EXPANSION CAPABILITIES

A fantastic bargain for only **$99.00** with the following features:

- 20 ma of RS 232 interface
- 64K addressing range
- Program control timers
- 1K off on board static memory
- Built in clock generator
- 64 Byte register
- Built-in priority interrupts
- Documentation

GENERAL PURPOSE COMPUTER POWER SUPPLY KIT

This power supply kit features a high frequency torroid transformer with switching transistors in order to save space and weight. 115V 60 cycle primary. The outputs with local regulators are +5V to 10A, in one amp increments,—5V at 1A, ± 12V at 1A regulators supplied 6 340T-5 supplied.

$79.00

UNIVERSAL 4K MEMORY BOARD KIT

$74.50

This memoryboard kit can be used with most microcomputers. Some of the outstanding features are:

32-2102-1 static RAM's, 16 address lines, 8 data lines in, 8 data lines out, all buffered. On board decoding for any 4 of 64 pages, standard 44 pin, .156″ buss.

4K F8 Basic $25.00

2708-8K EPROM $45.00
2522 STATIC SHIFT RE6 . . . . . . $ 1.95
2513 CHARACTER GEN . . . . . . $ 9.95
2518-HEX 32 BIT SR . . . . $ 3.50
2102-1 1024 BT RAM . . . . $ 1.49
5280-4K DYNAMIC RAM . . . . . $ 6.95
5202A UV PROM . . . . $ 6.95
MM5203 UV PROM. . . . . $ 6.95
1702A UV PROM . . . . $ 6.95
5204-4K PROM $10.95

MINIATURE MULTI-TURN TRIM POTS 100,500,1K,2K,5K,10K,100K,200K, 1 Meg. $.75 each . . . . 3/$2.00

MULTI-TURN TRIM POTS Similar to Bourns 3010 style 3/16″x5/8″x1-1/4″; 50, 100, 1K, 10K, 50K ohms $1.50 ea. . . . . 3/$4.00

LIGHT ACTIVATED SCR's TO-18, 200V 1A. . . . . $ 1.75

TRANSISTOR SPECIALS

2N3585 NPN Si TO-66 . . . . $ .95
2N3772 NPN Si TO-3 . . . . $ 1.60
2N456A PNP GE . . . . $ .75
2N4908 PNP Si TO-3 . . . . $ 1.00
2N6056 NPN Si TO-3 Darlington . $ 1.70
2N5086 PNP Si TO-92 . . . . 4/$ 1.00
2N4898 PNP TO-66 . . . . $ .60
2N404 PNP GE TO-5 . . . . . . 5/$ 1.00
2N3919 NPN Si TO-3 RF . . . . . . $ 1.50
MPSA 13 NPN Si TO-92 . . . . . 3/$ 1.00
2N3767 NPN Si TO-66 . . . . $ .70
2N2222 NPN Si TO-18 . . . . 5/$ 1.00
2N3055 NPN Si TO-3 . . . . $ .80
2N3904 NPN Si TO-92 . . . . 5/$ 1.00
2N3906 PNP Si TO-92 . . . . 5/$ 1.00
2N5296 NPN Si TO-220 . . . . $ .50
2N6109 PNP Si TO-220. . . . . $ .55
2N3638 PNP Si TO-5 . . . . 5/$ 1.00
2N651 7 NPN TO-92 Si . . . . 3/$ 1.00

C/MOS (DIODE CLAMPED)

74C02— .22	4015— .95	4029—1.10
74C10— .22	4016— .40	4030— .22
74C193-1.50	4017—1.05	4033—1.50
4001— .22	4018—1.00	4035—1.10
4002— .22	4019— .25	4042— .78
4006—1.20	4020—1.05	4047—2.00
4007— .22	4022— .95	4049— .40
4009— .42	4023— .22	4050— .40
4010— .42	4024— .75	4055—1.50
4011— .22	4025— .22	4066— .80
4012— .22	4026—1.25	4071— .22
4013— .40	4027— .40	4076—1.05
	4028— .88	

MCA-81 OPTICAL LIMIT SWITCH. $1.50

LED READOUTS

FND 500-.5″ C.C. $1.25
HP 7740-.3″ C.C. . . . . $ .95
MAN-7-.3″ C.A. . . . . $ .95
NS 33-3 dig. array . . . . $ .75
DL 747 . . . . $1.60

PRINTED CIRCUIT BOARD

4-1/2″x6-1/2″ SINGLE SIDED EPOXY BOARD 1/16″ thick, unetched $.50 ea. . . . . 5/$2.20

7 WATT LD-65 LASER DIODE IR $8.95

2N 3820 P FET . . . . $ .45
2N 5457 N FET . . . . $ .45
TIS 43 UJT . . . . $ .35
ER 900 TRIGGER DIODES . . 4/ $1.00
2N 6028 PROG. UJT . . . . $ .65
8 PIN DIP SOCKETS . . . . $ .24
14 PIN DIP SOCKETS . . . . $ .25
16 PIN DIP SOCKETS . . . . $ .28
18 PIN DIP SOCKETS . . . . $ .30
24 PIN DIP SOCKETS . . . . $ .40
40 PIN DIP SOCKETS . . . . $ .60

VERIPAX PC BOARD

This board is a 1/16″ single sided paper epoxy board, 4½″x6½″ DRILLED and ETCHED which will hold up to 21 single 14 pin IC's or 8, 16, or LSI DIP IC's with busses for power supply connector. $4.00

MV 5691 YELLOW-GREEN BIPOLAR LED $1.25
FP 100 PHOTO TRANS . . . . $ .50
RED, YELLOW, GREEN OR AMBER LARGE LED's . . . ea. $ .20
MOLEX PINS . . . . 100/$1.00
1000/$7.50
10 WATT ZENERS 3.9, 4.7, 5.6, 8.2, 10, 12. 15,18, 22, 100, 150 or 200V ea. $ .60
1 WATT ZENERS 4.7, 5.6. 18 OR 22V . . . . ea. $ .25
MC6860 MODEM CHIP. $9.95

Silicon Power Rectifiers

PRV	1A	3A	12A	50A	125A
100	.06	.14	.30	.80	3.70
200	.07	.20	.35	1.15	4.25
400	.09	.25	.50	1.40	6.50
600	.11	.30	.70	1.80	8.50
800	.15	.35	.90	2.30	10.50
1000	.20	.45	1.10	2.75	12.50

SILICON SOLAR CELLS

2¼″ diameter
.4V at 500 ma. $4.00 / .2V at 200 mils $2.00

REGULATED MODULAR POWER SUPPLIES

+–15 VDC AT 100ma 115VAC INPUT $27.95
5VDC AT 1A. 115VAC INPUT . . . $24.95
12 VDC AT .5 AMP $24.95
IN 4148 (IN914) . . . . 15/$1.00

Send 25¢ for our catalog featuring Transistors and Rectifiers
145 Hampshire St., Cambridge, Mass.

TANTULUM CAPACITORS

.22UF 35V	5/$1.00	6.8UF 35V	3/$1.00
.47UF 35V	5/$1.00	22UF 35V	$ .40
.68UF 35V	5/$1.00	30UF 6V	$ .40
1UF 35V	5/$1.00	33UF 35V	$ .40
2.20UF 20V	5/$1.00	47UF 20V	5/$1.00
3.3UF 35V	4/$1.00	100UF 35V	$ .35
4.7UF 35V	4/$1.00	150UF 15V	$ .50

M7001 ALARM CLOCK CHIP. . . . $5.75

NATIONAL MOS DEVICES

MM1402— 1.75	MM5057— 2.25
MM1403— 1.75	MM5058— 2.75
MM1404— 1.75	MM5060— 2.75
MM5013— 2.50	MM5061— 2.50
MM5016— 2.50	MM5555— 4.75
MM5017— 2.70	MM5556— 4.75
MM5055— 2.25	MM5210— 1.95
MM5056— 2.25	MM5260— 1.75

#24, EIGHT CONDUCTOR SPECTRA FLAT CABLE 10′/$1.50 100′/$13.50

#30 WIRE WRAP WIRE SINGLE STRAND 100′/$1.40

TTL IC SERIES

7400— .15	7445— .70	74151— .70
7401— .15	7446— .70	74153— .65
7402— .15	7447— .70	74154— 1.10
7403— .15	7448— .70	74155— .70
7404— .20	7450— .20	74157— .70
7405— .20	7472— .33	74161— .85
7406— .25	7473— .35	74163— .80
7407— .25	7474— .35	74164— .95
7408— .25	7475— .49	74165—1.05
7409— .21	7476— .35	74173—1.40
7410— .15	7480— .35	74174— .95
7411— .20	7483— .70	74175— .92
7412— .20	7485— .88	74176— .75
7413— .45	7486— .30	74177— .79
7414— .70	7489—1.85	74180 .70
7416— .25	7490— .45	74181—2.10
7417— .25	7491— .70	74190—1.20
7420— .20	7492— .50	74191—1.20
7425— .28	7493— .45	74192— .85
7426— .25	7494— .70	74193— .85
7427— .30	7495— .70	74194— .85
7430— .20	7496— .70	74195— .75
7432— .25	74107— .32	74196— .88
7437— .25	74121— .38	74257— 1.25
7438— .25	74123— .65	74279— .90
7440— .16	74125— .40	75324— 1.75
7441— .85	74126— .40	75491— .65
7442— .52	74132— .82	75492— .65

MINIATURE DIP SWITCHES
CTS-206-4 Four SPST switches in one minidip package. . . . $1.75
CTS-206-8 Eight SPST switches in a 16 pin DIP package. $1.95

5-8V SPST Miniature reed, relay., normally open, 330 Ohm coil resistance. $.75, 3/$2.00

ALCO MINIATURE TOGGLE SWITCHES
MTA 106 SPDT $1.20
MTA 206 DPDT $1.70

Full Wave Bridges

PRV	2A	6A	25A
200	.75	1.25	2.00
400	.95	1.50	3.00
600	1.20	1.75	4.00

SANKEN AUDIO POWER AMPS

Si 1010 G 10 WATTS. . . . . $ 7.95
Si 1020 G 20 WATTS. $15.95
Si 1050 G 50 WATTS. $27.95

CCD 110 LINEAR 256 XI BIT SELF SCANNING CHARGED COUPLED DEVICE $65.00

CCD 201 – 100 x 100 CHARGE COUPLED DEVICE $99.00

LINEAR CIRCUITS

LM307 —Op. Amp. . . . . $ .30
LM 309K 5V 1A REGULATOR . . $ .95
723 – 40 + 40VV REGULATOR . . $ .50
301/748-Hi Per. Op. Amp. . . . . . $ .31
320T 5,12,15, or 24V NEG REG . . $1.25
709C – Op. Amp. . . . . $ .31
78MGS-Pos 5 to 30V, 500 ma reg. $1.35
79MGS-Neg 5 to 30V, 500 ma reg. $1.35
CA 3047 Hi Pef. Op. Amp. . . . . .$ .95
340T 5, 6, 8, 12, 15, 18, 24V POS REG. TO-220 . . . . $1.10
741A or 741C OP AMP . . . . $ .31
LM 308 Oper. Amp., Low Power . . $ .95
747 – DUAL 741 . . . . $ .65
556 – DUAL TIMER $1.00
1456 – Oper. Amp. . . . . $ .95
LM 3900 – QUAD OP. AMP . . . . $ .49
LM 324 – QUAD 741 . . . . $1.50
560 – PHASE LOCK LOOP $2.00
561 – PHASE LOCK LOOP $2.00
565 – PHASE LOCK LOOP $1.25
566 FUNCTION GEN. $1.65
567 – TONE DECODER $1.50
LM 1310N FM STEREO DEMOD. . $2.75
8038 IC VOLTAGE CONT. OSC. . $3.90
LM 370 – AGC SQUELCH AMP. . $1.15
555 – 2μs – 2 HR. TIMER. . . . . . $ .45
553 QUAD TIMER . . . . $ .45
IL5 (MCT-2) OPTO-ISOLATOR . . $ .80
1458 DUAL OP AMP. . . . . $ .60
LM 380 – 2W AUDIO AMP. . . . . $ .95
LM 377 – 2W Stereo Audio Amp. . $2.50
LM 381 – STEREO PREAMP. . . . $1.50
LM 382 – DUAL AUDIO PREAMP $1.50
LM 311 – HI PER. COMPARATOR $ .90
LM 319 – Dual Hi Speed Comp. . . $1.25
LM 339 – QUAD COMPARATOR $1.50

TRIACS / SCR'S

PRV	TRIACS 1A	TRIACS 10A	TRIACS 25A	SCR'S 1.5A	SCR'S 6A	SCR'S 35A
100	.40	.70	1.30	.40	.50	1.20
200	.70	1.10	1.75	.60	.70	1.60
400	1.10	1.60	2.60	1.00	1.20	2.20
600	1.70	2.30	3.60		1.50	3.00

Terms: FOB Cambridge, Mass. Send Check or Money Order. Include Postage, Minimum Order $5.00, COD'S $20.00

SOLID STATE SALES
P.O. BOX 74B
SOMERVILLE, MASS. 02143 TEL. (617) 547-4005

WE SHIP OVER 95% OF OUR ORDERS THE DAY WE RECEIVE THEM

Reader Service

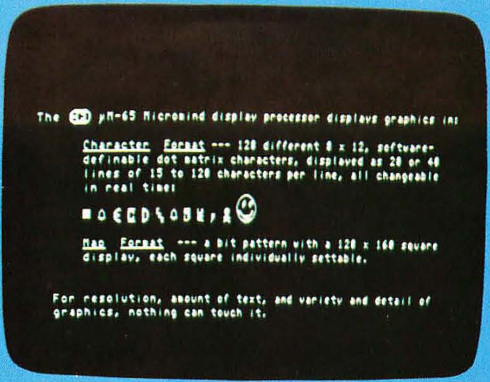
To get further information on the products advertised in BYTE, fill out the reader service card with your name and address. Then circle the appropriate numbers for the advertisers you select from this list. Add a 9 cent stamp to the card, then drop it in the mail. Not only do you gain information, but our advertisers are encouraged to use the marketplace provided by BYTE. This helps us bring you a bigger BYTE.

***Reader service inquiries not solicited. Correspond directly with company.**

BOMB— BYTE's Ongoing Monitor Box

Bombed . . .

The results of the BOMB tabulation for the month of January have now been prepared by Greg Spitzfaden. First prize winner in the contest was Dr Kenneth B Welles, for his article "Build This Economy Floppy Disk Interface." Second place winner was "The Digital Cassette Subsystem, Part 1" by Ira Rampil and Jack Breimier. First place winner Welles receives a $100 bonus check, and second place winners Rampil and Breimier receive a bonus of $50.■



Key Into Maxi-Power @ Micro-Price

Micromind is an incredibly flexible, complete and expandable, hardware/software, general purpose computer system. You won't outgrow it.

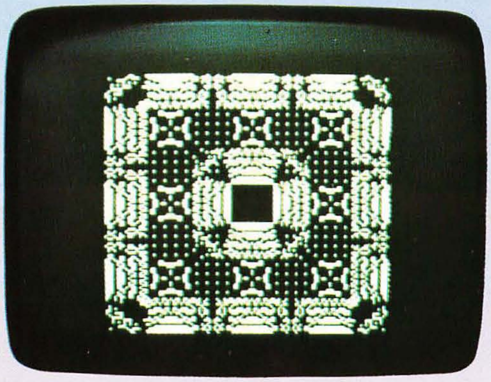
Hardware includes an 80 key, software-definable keyboard, I/O interface board, 6500A-series microprocessor (powerful enough for advanced computing), a high-detail graphics and character display processor, power supply, rf modulator, and connections for up to 4 tape recorders plus TV or monitor. An interconnect bus permits 15 additional microprocessors, parallel processing and vastly increased computing power.

System software—including ECD's own notsoBASIC high level language, on advanced error-correcting tape cassettes—provides a word processing editor, a powerful assembler, a debugger, a file system, graphic routines, and peripheral handlers. We also include dynamic graphic games: Animated Spacewar and Life.

ECD's standard Micromind μM-65 supplies 8K bytes of memory. Additional 32K byte expansion boards and a mapping option give Micromind expandable access to 64 Megabytes. Utilizing software-controlled I/O channels, Micromind's advanced encoding techniques load data from ordinary tape recorders at 3200 bits per second.







Micromind comes to you ready-to-use, factory assembled and fully tested. Among microcomputers, it has the largest memory capacity and the fastest storage. You're looking at the work of the finest display processor on the market. You won't find a microcomputer with a more powerful CPU. You won't find a computer with a more flexible keyboard. You won't find anything to touch it at **$987.54.**



So, quit the kluge scene and key into Micromind. You'll be a main frame performer, with all the comforts of home. We're not fooling...this is the cat's μ!

ECD CORP.
196 Broadway, Cambridge, Mass. 02139
(617) 661-4400

ECD

Name ______
Address ______
City/State ______ Zip ______
☐ Fantastic! Check enclosed: **$987.54.** Shipping paid by ECD
☐ BankAmericard ☐ Master Charge Mass. Resident add 5% Sales Tax
\# ______ Expiration Date ______
Signature ______
☐ Send me your brochure.
b

Actual unretouched photographs.

/ability.

It Comes Naturally With The Altair™ 8800b

The Altair 8800b from MITS: the second generation design of the microcomputer that started it all. The mainframe that has the abilities everyone is demanding from microcomputers today:

Expand-ability:

The Altair 8800b power supply and one-piece, 18-slot motherboard allow efficient and easy expandability for memory and I/O options. All Altair PC boards are designed to give you maximum capability/lowest power usage possible per board. This means that for each slot used you get more features and require less power, than with any of the "off-brand" Altair-bus-compatible boards.

Whether you buy an entire system up front or choose to expand gradually, it's easy to get the configuration you need with the complete family of Altair peripheral equipment, including floppy disk, line printer, audio cassette record interface, A/D converter, PROM programmer, serial and parallel I/O boards, choice of four different memory boards and many others.

Reli-ability:

The unique design features of the Altair 8800b, which have set the standard for the microcomputer industry, make it the most reliable unit of its kind. The Altair 100-pin bus, the now-standard design used by many imitators, has been "standard" all along at MITS. The unique Front Panel Interface Board on the Altair 8800b isolates and filters front panel noise before it can be transmitted to the bus. The all-new CPU board utilizes the 8080A microprocessor, Intel 8224 clock generator and 8216 bus drivers.

Flex-ability:

Meeting the diversified demands of an ever-increasing microprocessor market requires flexibility: not just hardware flexibility but software flexibility as well. MITS software, including the innovative Altair BASIC language, allows the full potential of the Altair 8800b computer to be realized.

8K ALTAIR BASIC has facilities for variable length strings with LEFT$, RIGHT$, and MID$ functions, a concatenation operator, and VAL AND STR$ functions to convert between strings and numbers.

Extended ALTAIR BASIC allows integer, single and double precision variables, automatic line numbering and renumbering, user-defined string functions, PRINT USING for formatted output and a powerful EDIT command for editing program files during or after entry. Extended statements and commands include IF . . . THEN . . . ELSE, LIST and DELETE program lines, SWAP variables and Trace On and Off for debugging.

Disk ALTAIR BASIC has all the features of Extended BASIC with the additional capability to maintain sequential and random access disk files. Utilities are provided for formatting disks and printing directories.

In all versions of ALTAIR BASIC you get the ease and efficiency of BASIC for the solution of real world problems.

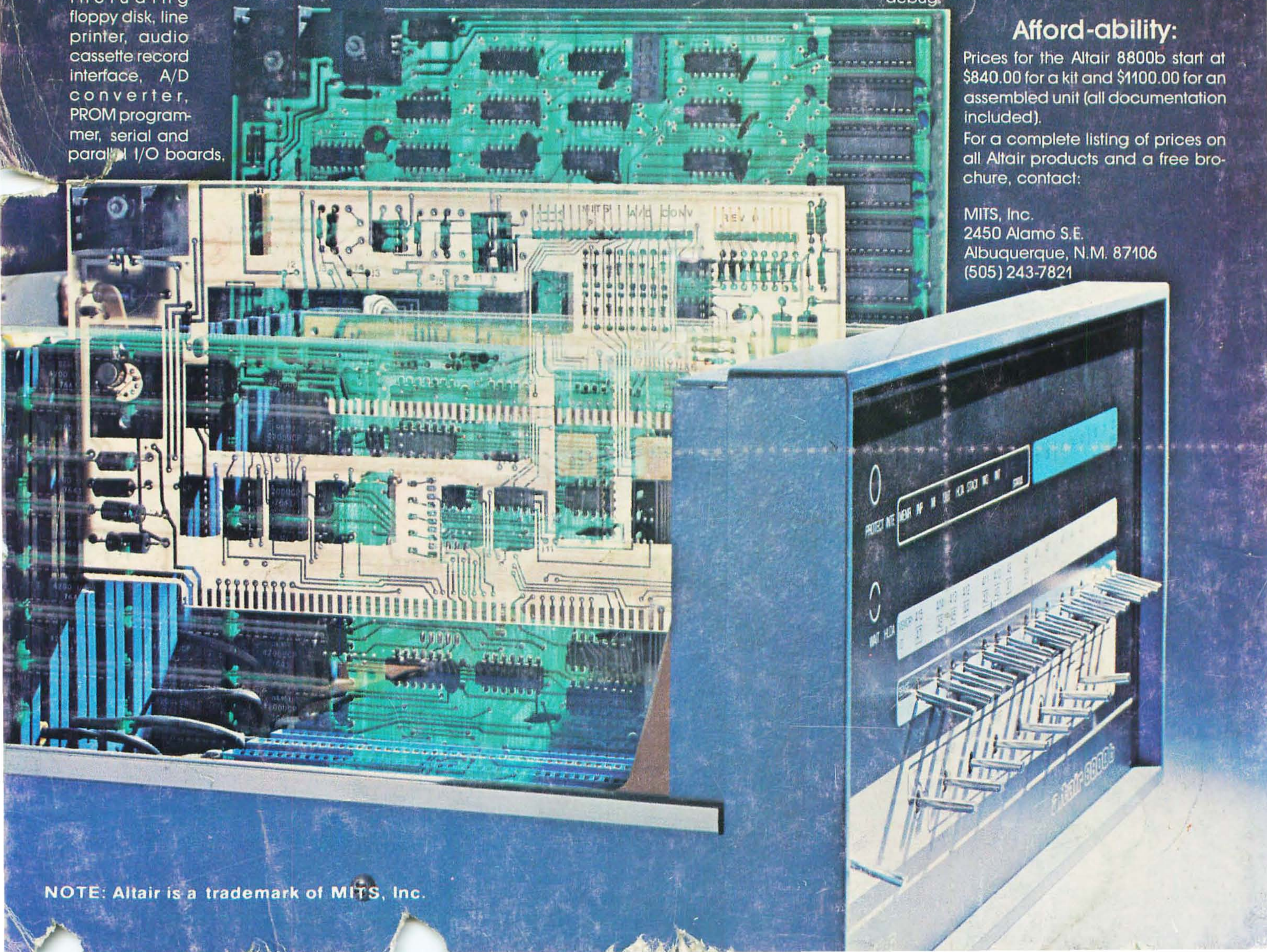
Package II, an assembly language development system for the Altair 8800b, includes system monitor, text editor, assembler and debug.

Afford-ability:

Prices for the Altair 8800b start at $840.00 for a kit and $1100.00 for an assembled unit (all documentation included).

For a complete listing of prices on all Altair products and a free brochure, contact:

MITS, Inc.
2450 Alamo S.E.
Albuquerque, N.M. 87106
(505) 243-7821



NOTE: Altair is a trademark of MITS, Inc.